ESD-TR-77-143

MTR-3345
Vol. I

PROGRAMMING LANGUAGES, STANDARDS, USE, AND SELECTION: AIR FORCE AND OTHER GOVERNMENT AGENCIES

VOLUME 1, ANALYSIS

OCTOBER 1977

Prepared for

DEPUTY FOR COMMAND AND MANAGEMENT SYSTEMS
ELECTRONIC SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
Hanscom Air Force Base, Bedford, Massachusetts

Project No. 572A
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract No. F19628-76-C-0001

REVIEW AND APPROVAL

This technical report has been reviewed and is approved for publication.

John C. Mott-Smith

JOHN C. MOTT-SMITH
Project Officer

John T. Holland

JOHN T. HOLLAND, Lt Col, USAF
Chief, Techniques Engineering Division

FOR THE COMMANDER

Frank J. Emma

FRANK J. EMMA, Colonel, USAF
Director, Computer Systems Engineering
Deputy for Command & Management Systems

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER ESD-TR-77-143	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) PROGRAMMING LANGUAGES, STANDARDS, USE, AND SELECTION: AIR FORCE AND OTHER GOVERNMENT AGENCIES VOLUME I, ANALYSIS		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER MTR-3345, Vol. I
7. AUTHOR(s) P. L. Loring, E. A. Lamagna and L. J. La Padula		8. CONTRACT OR GRANT NUMBER(s) F19628-76-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS The MITRE Corporation Box 208 Bedford, MA 01730		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project No. 572A
11. CONTROLLING OFFICE NAME AND ADDRESS Deputy for Command and Management Systems Electronic Systems Division, AFSC Hanscom Air Force Base, MA 01731		12. REPORT DATE OCTOBER 1977 13. NUMBER OF PAGES 136
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

HIGH ORDER LANGUAGE
PROGRAMMING LANGUAGE
SOFTWARE ACQUISITION
STANDARDIZATION

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Information related to computer programming languages was collected about U.S. Air Force systems and about activities at other government agencies during Phase II of the Air Force Systems Command High Order Language Standardization Program. An understanding of this experience in procuring, developing, and operating weapon and defense systems involving embedded computer resources is required in order to

(over)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (continued)

assist decision-making regarding standardization of programming languages for Air Force use. Facts were reported on 64 U.S. Air Force systems and on related programs at NASA, the FAA, and other government agencies. An analysis of this data is presented in Volume I and system summaries are presented in Volume II.

ACKNOWLEDGMENTS

This report has been prepared by The MITRE Corporation under Project 572A. The contract is sponsored by the Electronic Systems Division, Air Force Systems Command, Hanscom Air Force Base, Massachusetts.

This report has been produced by the AFSC High Order Language (HOL) Standardization Program Task in support of the Information Systems Technology and Applications Office (more recently named Computer Systems Engineering) of Electronic Systems Division (ESD) as part of Air Force Project 2237.

Air Force Systems Command (AFSC) had directed ISTAO in late 1974 to investigate HOL standardization because of increasing costs associated with software production and maintenance. As part of the resulting program, a knowledge baseline was assembled. The information in this report is a principal component of this knowledge baseline; it represents the contribution of Air Force and other government agencies to Phase II of the HOL Standardization Program.

Our thanks to many MITRE, Air Force, NASA, and other government personnel who contributed their time and their knowledge of specific programs. Without their help this report could not have been written.

We would also like to thank D. Kirshner, M. Miceli, and J. Mata of MITRE who helped assemble the data reported herein and who contributed greatly to the quality of the final product.

Special thanks to John Mott-Smith ESD/MCIT whose assistance in coordinating the inputs and verifying the analysis contributed significantly to the final product.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Concl.)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I

INTRODUCTION

During Phase II of the Air Force Systems Command (AFSC) High Order Language Standardization Program, information related to computer programming languages was collected about Air Force systems and about activities at other government agencies. An understanding of this experience in procuring, developing, and operating weapon and defense systems involving computer software is required in order to assist decision-making regarding standardization of programming languages for Air Force use.

Volume I includes an analysis of the data on Air Force experience (Sections II through IV). Non-Air Force experience (Appendix I) and other standardization experience (Appendix II) serve to corroborate Air Force requirements and to support the conclusions of this study (Section V). In addition, descriptions of the computer architectures used in the Air Force (Appendix III) and Software Engineering experience (Appendix IV) are reported. Volume II contains summaries of each of the Air Force systems surveyed (Section II) and the activities of the non-Air Force agencies which contributed to this data collection effort (Section III).

Air Force systems surveyed fall into the following principal application areas:

Automatic Test Equipment (ATE)
Communications (COMM)
Command and Control (CC)
*Information Management (IM)
*Navigation (NAV)
Operational Flight Programs (OFP)
Range Operations (RO)
Simulator and Trainer (ST)
*Support Systems (SUP)
*Surveillance and Warning (SW)

These application areas are described in Section II (Definition of Application Areas). They include the six categories used during Phase I of this program [LAPA76] but have been increased (by those marked with *) to reflect functional distinctions made possible by

*Newly created application areas.

the greater detail and scope of data collected during Phase II. These areas are delineated on the basis of operating environment and functional requirements of the application software; they do not necessarily reflect organizational responsibilities.

Background

In October 1974, the Information Systems Technology Applications Office (ISTAO)**, ESD/MCI, was directed by Air Force Systems Command (AFSC) to evaluate high order language (HOL) standardization. MITRE, under Air Force Project 2237, was tasked to support ISTAO in this effort.

The long-range objectives of the HOL Standardization Program are to develop recommendations on HOL standardization and to provide a standardization policy and implementation plan. Starting in FY75, MITRE was tasked to assist in determining Air Force software requirements in six application areas, developing information on Air Force language selection practices and principles, and collecting and annotating existing and in-progress studies on HOL evaluation and selection. The first year's task summary gives an overview and task description for Phase I of the program.

Together with the Phase I products, primarily [LAPA76], this Phase II report completes the knowledge baseline developed by the HOL Standardization Program. Efforts next year will concentrate on formulating a draft Air Force policy and implementation plan for HOL standardization which will be coordinated among participating Air Force organizations.

Approach

During Phase II, an outline for data collection was prepared and transmitted to:

Air Training Command (ATC)
Air Force Avionics Laboratory (AFAL)*
Armament Development and Test Center (ADTC)*
Strategic Air Command (SAC/ADXRM and SAC/DOK)*
Military Airlift Command (MAC)*
Tactical Air Command (TAC)*
Naval Air Training Center (NATC)
Aerospace Corporation

**Now called Computer Systems Engineering (CSE).

National Aeronautics and Space Administration (NASA, including ten Space Centers and other facilities)*
Army Computer Systems Command
Army Electronics Command
NAVAIR Systems Command
NAVAL Electronics Laboratory Center
Defense ARPA
National Bureau of Standards
RAND Corporation
Charles Stark Draper Lab*
Aeronautical Systems Division (AFSC/ASD)*
Space and Missile Systems Division (AFSC/SAMSO)*

Organizations with a * responded with data at differing levels of detail. In addition, MITRE and ESD systems were surveyed directly. A memorandum was sent to all MITRE project leaders and all ESD SPOs responsible for acquisition programs involving software development. The questionnaire attached to this memo was in two parts; the first asked for readily available information and the second asked in which areas could more detailed data be obtained. MITRE personnel, where possible, and ESD personnel in other cases, were subsequently interviewed to obtain this detailed data. A draft MITRE working paper was prepared and circulated for comments from ESD and MITRE project personnel, leading to a final version of the working paper which provides about half the data reported in Volume II of this report.

Other HOL Standardization Activities

In January 1975, the Director of Defense Research and Engineering established a working group composed of representatives of each of the military departments to study the possibility of defining a minimal set of standard military higher order language(s) for non-COBOL and non-FORTRAN applications. This working group currently functions as an adjunct to the DoD Management Steering Committee for Embedded Computer Resources established by DoDD 5000.29.

To date, this working group has produced three successive versions of language requirements, a "STRAWMAN," a "WOODENMAN," and a "TINMAN" [FISH76]. MITRE and ISTAO are among the many government, industry, and academic organizations and individuals that have critiqued this evolving list. AFSC/XRF has collected all Air Force inputs and has represented the Air Force on this working group. The resulting Air Force position on computer programming language requirements depends heavily on the findings of the AFSC HOL Standardization Program.

DoDD 5000.29, April 1976, [DODD76A] requires use of a programming language from a list of interim standards until the DoD standard language(s) is developed or selected for major weapon and defense systems; DoDI 5000.31, November 1976, [DODI76] identifies the allowed interim languages and Control Agent for each. Other languages are permitted if none of the approved HOLS are "cost effective or technically practical over the system life cycle." The Air Force is preparing to comply with the requirements of DoDI 5000.31. The AFSC HOL Standardization Program predates the DoD study and, because of the similarity of goals, has served as a major source of information on Air Force requirements to DoD.

The list of DoD interim standard High Order Programming Languages and their control agents [DODI76] is:

COBOL - Office of the Assistant Secretary of Defense (Comptroller)
FORTRAN - Office of the Assistant Secretary of Defense (Comptroller)
CMS-2 - Department of the Navy
SPL/I - Department of the Navy
TACPOL - Department of the Army
JOVIAL (J3 and J73) - Department of the Air Force

SECTION II

AIR FORCE EXPERIENCE

Air Force systems, for purposes of this report, have been grouped into ten application areas according to functional similarities; the major characteristics of systems in these areas are described in the first subsection. The subsection that follows displays the data reported in Volume II in tabular form in order to highlight significant facts and provide a basis for subsequent analysis.

DEFINITION OF APPLICATION AREAS

This section provides a basic description of each of the ten application areas identified in the introduction and listed in Figure 1. The principal functions performed by and the requirements placed on the software in each area are outlined, with emphasis being given to the application software.

The application area classification scheme employed in this report is an extension and modification of that used in Phase I of the AFSC HOL Standardization Program [LAPA76]. The new scheme was necessitated by the broader diversity of systems investigated in Phase II of the program and the desire to have a more functionally descriptive, yet flexible, classification method.

In the new classification scheme, each system is assigned the application area designation which best describes the primary mission of the system. Descriptors giving additional information, such as the nature of a system's mission or its deployment, are optionally used in conjunction with the principal application area. Figure 1 lists the most frequently used descriptors for each of the principal application areas.

Many of the system data summaries appearing in Volume II of this report include both "reported" and "functional" application area designations. For such systems, the "reported" application area represents the information provided by the organization reporting data on the system and, in most cases, is based on the classification scheme used in Phase I of the HOL Standardization Program. The "functional" application area is the designation which has been assigned to a system within the new classification scheme. The discussion and analysis in this volume is geared toward the "functional" area designation.

APPLICATION AREA	DESCRIPTORS	
Automatic Test Equipment (ATE)		
Command and Control (CC)	Mission strategic tactical air defense satellite	Deployment ground airborne
Communications (COMM)	Type transmission media terminals message switches networks	
Information Management (IM)		
Navigation (NAV)	Type air traffic control navigation via satellites ground support for avionics OFPs	
Operational Flight Programs (OFP)	Mission avionics electronic warfare space missiles	
Range Operations (RO)	Mission avionics missiles	
Simulator and Trainer (ST)	For avionics flight crews air traffic control command and control	
Support Systems (SUP)	For intelligence operational flight programs command and control missiles	
Surveillance and Warning (SW)		

Figure 1. Descriptors Used with Application Areas

The remainder of this section is devoted to a brief overview of the ten principal application areas.

Automatic Test Equipment (ATE)

Automatic test equipment generally performs testing of mechanical, electro-mechanical, or electronic subassemblies of systems. Emphasis in systems reported in Phases I and II of this Program is on electronic subassemblies (e.g., printed circuit boards). ATE, therefore, includes programs which operate in a special test environment designed to diagnose faulty electronic equipment in weapon and defense systems. Analog, digital, and hybrid tests are used to identify, diagnose, and isolate faults in electronic units. The principal functions performed by the application software of ATE systems include test pattern generation, performance monitoring, and analysis of test results.

ATE systems are generally procured with new weapon and defense systems to fulfill testing requirements. ATE field systems consist of analog and digital test equipment and either a minicomputer or a network of mini or microprocessors. Both commercially available computers and hardware militarized to the extent of withstanding transportation and storage effects are employed. Software development is sometimes performed on larger general-purpose computers, and sometimes is done on the ATE mini. Software development costs for ATE systems are of great importance due to the large number of programs which must be developed for each new ATE system; efficiency of the resulting object code is less critical.

Command and Control (CC)

Command and control systems are usually large ground-based systems which collect live situation information, maintain sizable supporting data bases, and direct actions of and supply information to offensive and defensive systems. Airborne CC includes the E-3A, Airborne Warning and Control System. Since there is a good deal of information flow to and from CC systems, most CC systems entail a good deal of communications processing.

Strategic, tactical, and air defense are three major types of CC systems. These designations serve as mission descriptors in the application area classification scheme employed in this report. Deployment descriptors (ground and airborne) are also used.

The principal functional requirements of such CC systems are as follows:

1. Information development

   - surveillance
   - detection
   - identification
   - tracking

2. Decision aids

   - situation monitoring
   - force control
   - evaluation of alternatives, making recommendations
   - operations monitoring

3. Planning

   - collecting resource status information
   - matching resources and requirements
   - scheduling activities (e.g., airlift and tactical missions)

4. System test and training functions

There are also CC systems for the command and control of satellites which perform the major functions of vehicle control, orbital calculations, monitoring, and information transmission and analysis.

CC systems are acquired infrequently; each lasts about 20 years. They incorporate both commercially available and militarized computers. The central processors of CC systems are generally large-scale computers, while minicomputers are often used as communications or peripheral or subsystem controllers. CC systems perform many of their functions in real time, in which case object code efficiency is of great importance.

Communications (COMM)

The communications application area includes programs which assist in or perform the transmittal of information through a communications channel. The following basic types of COMM systems, which serve as descriptors for the application area, can be identified:

1. Transmission media - communication channels for the transmission of messages from source to destination.

2. Terminals - interfaces between communication systems and users.

3. Message switches - single processors which perform such message handling functions as coding, format conversion, routing, and link protocol.

4. Networks - networks of processors which together perform the message handling functions of switches.

Many of the systems surveyed in this report, including most of those designated as command and control, involve a good deal of communications processing in the performance of their missions. A system is designated as COMM only if its principal functions are involved with communication processing; those systems which perform incidental communication processing in support of another primary mission have been given other application area designations.

The COMM systems reported here run on both commercially available and militarized computers, depending on system requirements. Real-time responsiveness and high reliability are critical requirements of COMM systems.

Information Management (IM)

Information management systems are concerned with the storage and retrieval of information into and from a data base. The principal functions performed by IM systems are file maintenance, responding to user queries and requests, report production, and general data processing.

Information management systems operate in both real-time and batch mode, and run on commercially available equipment.

Navigation (NAV)

Navigation systems are used to plot, ascertain, or direct the course of aircraft. The types of NAV systems investigated, which serve as descriptors for this application area, include systems for air traffic control, navigation via satellites, and ground support for avionics operational flight programs.

Common functions performed by navigation systems include providing signals for position estimation, beacon processing, tracking, and display processing. NAV systems operate in real-time or near-real-time. The systems investigated run on minicomputers of both the militarized and commercially available variety.

Operational Flight Programs (OFP)

Operational flight programs run in real-time on militarized airborne minicomputers which are supported by ground-based general purpose computers. These programs are used in airborne weapons systems, such as fighters and bombers, and defensive space and missile systems. OFP missions, which serve as descriptors in the application area classification scheme, include avionics, missile, and space-related missions, and electronic warfare.

Avionics software accomplishes a variety of tasks and serves to integrate avionics systems. Functions performed by avionics OFP [FALK76] include display processing; vehicle navigation, guidance, and control; weapons delivery and launch/deployment control; cargo airdrop; stores management; radar signal processing and tracking; and electronic warfare and countermeasures (viz., threat detection, threat identification, threat prioritization, jammer control, power management).

Space and missile system in-flight (mission-oriented) application software functions [CALL75] include vehicle navigation, guidance, and control; surveillance; reconnaissance; data collection and transmission; weapon delivery (missile systems only); command and control; life support; and experiment support.

Efficiency and reliability are major concerns of operational flight programs. Maintainability and transferability, although important, are of secondary importance. Maintainability is a concern as it affects life cycle costs.

OFP control real-time operations and must respond to real-time inputs. Furthermore, airborne computers are subject to severe size, weight, and power constraints which limit memory and computer size.

Hence, operational flight software must be efficient with respect to execution time and memory utilization. Additionally, spaceborne computers must have long-life, be self-repairing, and withstand a severe environment.

Operational flight software is developed on ground-based computers, generally commercially available equipment, which are used to assemble, test, and link computer programs for operational use. A wide variety of support software is usually required for the development and test of OFPs. In addition to operational program development, functions performed by ground-based application software include targeting, mission simulation, radio control guidance, data reduction from analog sources, and range support.

Range Operations (RO)*

The vast majority of range operation programs support missile test firing activities by performing such functions as trajectory calculations, impact prediction, and weather surveillance. Data on a range operation system for avionics is also reported. Descriptors for this application area indicate whether an RO system supports an avionics or missile-related mission.

The principal functions performed by RO systems in support of range testing and safety for various offensive and defensive missile weapon systems may be divided as follows:

1. Field systems for radar management and meteorological data processing.

2. Range safety operations including real-time tracking, impact prediction, and mission simulation.

3. Support systems for training, scenario preparation, post-flight analysis, data base management, and computer utilities.

Range operation programs run on both large and medium-size ground-based general-purpose computers. Most range operation systems perform substantial amounts of scientific computation. The reliability of an RO system is of paramount importance for mission execution and safety. RO systems operate in real-time during test

---

*This category was called Range Support (RS) in Phase I [LAPA76].

execution; auxiliary support systems perform ordinary data processing functions which need not be done in real-time. Acquisition of new computers occurs infrequently; new requirements are usually satisfied by the development of new software for existing equipment.

Simulator and Trainer (ST)

Simulator and trainer programs are designed to assist in the operation of training devices. The vast majority of ST systems deployed by the Air Force support avionics flight crew training for airborne weapon system operation. Data for this report have been collected on an ST system for air traffic control and on another for command and control. The mission supported by an ST system (e.g., avionics, air traffic control, command and control) serves as a descriptor in the application area classification scheme used in this report.

The principal software functions performed by simulator and trainer systems may be divided as follows:

1. exercise preparation - generation of training scenarios

2. exercise conduct - display generation, responding to switch actions of trainees

3. data analysis

Minicomputers are generally employed in ST systems. There is a need for efficient object code in exercise conduct routines which perform in real-time a variety of functions simulating the aggregate behavior of an operational system.

Support Systems (SUP)

Support systems supply information processing capabilities to other missions. This support usually takes the form of providing facilities for scientific data processing, data base management, and reporting. All of the support systems investigated here operate out of ground-based centers and employ commercially available large-scale computers. No requirement for real-time or compile-time efficiency has been identified.

In the classification scheme used in this paper, descriptors for the SUP area identify the nature of the mission for which support is provided (e.g., intelligence, operational flight programs, command and control, missiles).

Surveillance and Warning (SW)

The principal functions performed by surveillance and warning systems are air surveillance, detection, identification, and tracking. These are the information development functions associated with the command and control application area. SW systems are distinguished from CC systems in that they perform either limited or no decision-making functions, and instead forward information to CC systems for such processing when appropriate.

SW systems are driven by real-time external events (i.e., radar inputs) and therefore must be supported by efficient object code. All of the SW systems reported are ground-based and run on commercially available data processing equipment.

DATA SUMMARY

The seven tables included at the end of this section present information on the sixty-four Air Force systems described in Volume II, System Summaries. The first table serves as an introductory reference, listing the systems alphabetically and classifying them according to the ten major application areas described in Volume I. The remaining tables list the systems at the top, grouped by major application area. The data presented in these tables is useful in determining trends in system software acquisitions; the data displayed forms the basis for the subsequent analysis. Information regarding compiler availability in Table V was obtained from the Auerbach Computer Technology Reports [AUER76], Datapro 70 [DATA76], and Computer Review [GML 76]. All other information for these tables was obtained from Volume II.

The sixty-four systems, the expansion of their acronyms, and their application area designation are:

- ACTS (Automated Communications Test Software) for FLTSATCOM (Fleet Satellite Communications) - ATE

- ADTC/TSX (Armament Development and Test Center) Systems - RO, missiles

- AFAL (Air Force Avionics Laboratory) Operational Flight Programs - OFP, avionics

- AFEES (Automated Armed Forces Entrance and Examination Station) - IM

- AFSATCOM I (Air Force Satellite Communications I) - COMM, transmission media (satellite)
- AFSATCOM II/III (Air Force Satellite Communications II/III) - COMM, transmission media (satellite)
- AFSCF (Air Force Satellite Control Facility) - CC, satellites
- ASTROS (Advanced Systematic Techniques for Reliable Operational Software) - RO
- ATEC (Automated Technical Control) - ATE
- B-1 Strategic Bomber - OFP, avionics including electronic warfare
- C-5 Cargo Transport Aircraft - OFP, avionics
- CCPDS (Command Center Processing and Display System) - CC, strategic, warning
- COBRA DANE - SW
- COMBAT GRANDE (Semiautomated Spanish Air Defense System) - CC, air defense
- CONUS OTH (Continental United States Over-the-Horizon Radar System) - SW
- CSDRO (Computer Services Division Range Operations) - RO, missiles
- DMSP Command and Control Support (Defense Meteorological Satellite Program) - SUP, Command and Control
- DMSP Ground Segment (Defense Meteorological Satellite Program) - CC, satellite
- DMSP Space Segment (Defense Meteorological Satellite Program) - OFP, satellite
- DS&A (Data Services and Analysis Program) - SUP, missiles
- E-3A (AWACS, Airborne Warning and Control System) - CC, airborne, tactical

- E-4 Block I (AABNCP-I, Advanced Airborne Command Post) - COMM, terminal

- E-4 Block II (AABNCP-II, Advanced Airborne Command Post) - IM, airborne, strategic command and control

- EF - 111A Tactical Jamming system - OFP, avionics, electronic warfare

- F-15 Air Superiority Fighter - OFP, avionics, including electronic warfare

- F-16 Lightweight Fighter - OFP, avionics

- GEODSS (Ground-Based Electro-Optical Deep Space Surveillance System) - SW

- GERTS Guidance System (General Electric Radio Tracking System) - RO, missiles

- IDHS (Intelligence Data Handling System) - IM, intelligence

- JSS (Joint Surveillance System) - SW

- JTIDS/ASIT (Joint Tactical Information Distribution System/Adaptable Surface Interface Terminal) - COMM, Terminal

- LORAN AN/ARN-101 (V) (Tactical Long Range Navigation) - OFP, avionics

- LORAN C/D Ground Chain (Tactical Long Range Radio Navigation, AN/TRN-38(V)) - NAV, ground support for avionics OFP

- MACIMS (Military Airlift Command Integrated Management System) - IM

- Minuteman III WS1334-M and WS133B Weapon System - OFP, missiles

- NAVSTAR GPS (Global Positioning System) - NAV, via satellites

- NORAD CMC Improvements (North American Air Defense Cheyenne Mountain Complex) - CC, air defense

- PACOM C4 (Pacific Command Command, Control, Computer, Communications) - CC, strategic and tactical

- PAVE PAWS (Phased Array Warning System) - SW

- PELSS (Precision Emitter Locater Strike System) - OFP, avionics

- RISS (Reconnaissance Intelligence Support System) - SUP, intelligence data gathering

- RTF (Remote Terminal Facility) - COMM, terminal

- SACCS/DTS (Strategic Air Command Automated Command Control System/Data Transmission Subsystem) - COMM, message switching

- SACCS/FMIS (SAC Automated Command Control System/Force Management Information System) - CC, strategic

- SACOPS (SAC Operational Planning System) - SUP, missile operational flight programs

- SATIN I (SACCS AUTODIN TTY Interface) - COMM, message switching

- SATIN IV (SAC Automated Total Information Network) - COMM, network

- SDS (Satellite Data Systems) - CC, satellite

- SK Satellite Control Systems - CC, satellite

- STEM (System Training and Exercise Module), Tactical Air Control System Improvements (TACSI) - ST, tactical command and control

- TACC AUTO (Tactical Air Control Center Automation) - Tactical Air Control System Improvements (TACSI) - CC, tactical

- TACS/TADS (Tactical Air Command System/Tactical Air Defense System) - CC, tactical

- TIPI (Tactical Information Processing and Interpretation) - CC, tactical - includes four segments which are individually represented:

TIPI-DC/SR (TIPI - Display Control/Storage and Retrieval)

TIPI-IAC (TIPI - Intelligence Analysis Center)

TIPI-II (TIPI - Imagery Interpretation Segment)

TIPI-TERPE (TIPI-Tactical Electronics Reconnaissance and Evaluation)

- TOSS (Terminal Oriented Support System) - COMM, message switching

- TRACALS - PIDP (Traffic Control and Landing Systems - Programmable Indicator Data Processor) - NAV, air traffic control

- TRACALS - VFR Control Tower (Traffic Control and Landing Systems, AN/GSN-T-3) - ST, air traffic control

- TRI-TAC/Combat Theater Communications (Joint Tactical Communications Program): Tactical Communications Control Facilities (TCCF) - COMM, network

- USAF TFWC Support (USAF Tactical Fighter Weapon Center) - RO, avionics

- Wild Weasel Fighter - OFP, avionics

- WWMCCS (World-Wide Military Command and Control System), especially AFWWMCCS - CC, strategic and tactical

- WWMCCS II (World-Wide Military Command and Control System) - SUP, avionics operational flight programs

Table I depicts the major application area of each of the sixty-four systems investigated in Volume II. Each system is classified into one of the application areas described in the previous section.

Table II lists each system and the Major Command or organization that reported the information on that system. Also shown is the status of each system.

Table III lists, by manufacturer and series, the principal hardware employed by the sixty-four reported systems. Since most systems use many computers, parentheses are used to identify major processing units in order to distinguish them from support hardware. The letter D is used to denote hardware that is used for software

development of a particular system. Systems marked "unknown" are still in the selection process or have not reported on their hardware.

Hardware is grouped into three categories; the first ten machines that appear at the top left-hand portion of Table III comprise the first category, the large-scale computers. The second category, the medium-scale computers, is comprised of twelve machines. The third category is a group of twenty commercial and militarized computers; this group of mini and microprocessors find small-scale use primarily in support of operational flight programs. Criteria for these classifications involves not only physical size, but memory capacity, word size, and processing capabilities. Refer to Appendix III for a complete listing of this third group of computers and descriptions of the twenty-two major machines.

Table IV lists the programming languages used to write application software in each of the sixty-four reported systems. Some systems make use of more than one language, but only one or two are used as the primary language for application programming; the rest are incidental in use, as indicated by parentheses. In cases where there is a question mark, the language or compiler is undecided because the acquisition program is still in the selection process.

Table V lists, by computer system, compilers that are offered by the computer maufacturer or owned by the Air Force. Parentheses are used to indicate compilers that are used in reported systems. In cases where a compiler(s) is offered, but not used, an assembler is employed. JOVIAL compilers for the Honeywell 6000 Series, the UNIVAC 1100 Series, and the CDC CYBER 70 Series are included in the maufacturer's software packages. All other JOVIAL compilers were developed specifically for and are owned by the Air Force; a J3 compiler for the Honeywell 6000 Series is also Air Force owned. This table lists computers which serve as host machines for compilers; it does not list cross-compilers or indicate target machines, e.g., airborne computers, which can execute code compiled on the host computer.

Table VI presents the criteria affecting the selection of programming languages in the sixty-four reported systems. The top rows indicate what agent, Air Force agency or contractor, selected the language and if the decision was discretionary or required. These six categories are:

1. Requirement: Air Force directive - an official Air Force requirement dictates the use of one or more specific

languages, for example AFR 300-10 requires use of JOVIAL (J3) for command and control applications.

2. Requirement: User/SPO - the using command or Program Office (SPO) requires one or more specific languages; this requirement is reflected in the Request for Proposal (RFP) package.

3. Requirement: User/SPO (class of language) - the using command or Program Office (SPO) requires use of one or more languages from a class of languages such as any high order language (HOL), a specified list of HOLs, or a combination of HOL and assembly language. This class of language is enumerated in the Request for Proposal (RFP) package.

4. Developer discretion: contractor - the RFP does not restrict the choice of language and the developing contractor selects the language.

5. Developer discretion: user - the RFP does not restrict the choice of language and the user, serving as a software developer, selects the language.

6. Developer discretion: Air Force/other organization - either the RFP does not restrict the choice of language or this is a planning/feasibility study; the software developer is an Air Force or other organization (such as MITRE) that is not the user and this organization selects the language.

The lower portion of Table VI is divided into thirteen categories that show the underlying reasons why a certain language was selected. Each category lists a factor, a characteristic, or quality of the system that influenced the language selection process. These thirteen categories are:

1. Overall system design - the soundness of the total system design, including hardware and software, is the major criterion for selecting a contractor; the programming language(s) is an integral part of the design.

2. Suitability for application - the language(s) selected was perceived to be well-suited to the system's functional requirements, such as data base handling or scientific computations.

3. Processing requirements - the language(s) was selected to meet the system's processing requirements, such as memory or timing constraints.

4. Hardware selection - the decision to use particular computer hardware was made for cost, performance, or availability reasons before the selection of languages and/or compilers. Languages for which translators were available with the hardware were chosen.

5. Off-the-shelf approach - Program Offices (SPOs) are reluctant to pay for development of new support software, especially compilers; bidding contractors are required to have operational compilers and/or assemblers. SPOs also frequently acquire hardware off-the-shelf but that is not included in Table VI.

6. Availability of compilers - the availability of a particular language compiler influenced the choice of language(s). Software development costs are reduced by choosing a system with an available compiler. Once hardware is selected, one of the available compilers is used, although an off-the-shelf compiler was not required.

7. Programmer training - choice of language is influenced by the expectation of reduced programmer training time and/or cost. Such savings are possible when programmers are already knowledgeable or proficient in a specific language or class of language or when a particular language is perceived to be easy to learn.

8. Maintainability/reliability - ease of maintaining a particular language or class of language influenced language selection.

9. Software transportability - plans to reuse application programs written in a high order language influence language choice. These programs are either being transferred from an existing system to one under development (build on existing investment) or between systems which are both under development (avoid duplication of effort). Software transportability minimizes the required reprogramming effort.

10. User experience - the system's user had previous experience with the programming language(s) selected. The user has confidence in the language's ability to do the job and may

also be associated with other systems using the same language. User experience may also manifest itself in other factors such as programmer training.

11. Standard language - a language implemented by many compilers and which is widely known, accepted, and supported is desired. In some cases programming begins before hardware is selected or available.

12. Reuse of compilers - plans to reuse compilers or other support software in order to minimize reprogramming and maintenance effort influence language selection.

13. Software engineering support - the suitability of the language to software engineering techniques or availability of SE support tools influenced language choice.

14. Unknown - no data was reported.

Table VII lists the agent, contractor, or Air Force organization that is responsible for developing and maintaining the software used in each of the sixty-four reported systems. The type of software developed or maintained is of two types, system/support and application; they are denoted by an X and a circle, respectively. Responsibility for development and maintenance falls into three categories, the contractor, the user, or another Air Force organization such as CCTC (Command and Control Technology Center), and CCPC (Communications Computer Processing Center), Air Force Communications Service, Tinker AFB. Systems with a mark in the WWMCCS row use WWMCCS hardware and software, and depend on CCTC for system/support software maintenance. If development or maintenance is listed as unknown, the system is either in planning stage and has not decided, or information was unavailable.

Table I

System Application Areas

	1 ACTS	2 ADTC/TSX	3 AFAL	4 AFEES	5 AFSATCOM I	6 AFSATCOM II/III	7 AFSCF	8 ASTROS	9 ATEC	10 B-1 BOMBER	11 C-5 CARGO	12 CCPDS	13 COBRA DANE	14 COMBAT GRANDE	15 CONUS OTH	16 CSDRO
AUTOMATIC TEST EQUIPMENT	X								X							
COMMAND AND CONTROL							X					X		X		
COMMUNICATIONS					X	X										
INFORMATION MANAGEMENT				X												
NAVIGATION																
OPERATIONAL FLIGHT PROGRAMS			X							X	X					
RANGE OPERATIONS		X						X								X
SIMULATOR AND TRAINER																
SUPPORT																
SURVEILLANCE AND WARNING													X		X	

	17 DMSP COMMAND SUPPORT	18 DMSP GROUND SEGMENT	19 DMSP SPACE SEGMENT	20 DS&A	21 E-3A AWACS	22 E-4 BLOCK I	23 E-4 BLOCK II	24 EF-111A	25 F-15	26 F-16	27 GEODSS	28 GERTS	29 IDHS	30 JSS	31 JTIDS/ASIT	32 LORAN AN/ARN-101
AUTOMATIC TEST EQUIPMENT																
COMMAND AND CONTROL		X			X											
COMMUNICATIONS						X									X	
INFORMATION MANAGEMENT							X						X			
NAVIGATION																
OPERATIONAL FLIGHT PROGRAMS			X					X	X	X						X
RANGE OPERATIONS												X				
SIMULATOR AND TRAINER																
SUPPORT	X			X												
SURVEILLANCE AND WARNING											X			X		

	33 LORAN c/d GROUND CHAIN	34 MACIMS	35 MINUTEMAN III	36 NAVSTAR GPS	37 NORAD CMC	38 PACOM C4	39 PAVE PAWS	40 PELSS	41 RISS	42 RTF	43 SACCS/DTS	44 SACCS/FMIS	45 SACOPS	46 SATIN I	47 SATIN IV	48 SDS
AUTOMATIC TEST EQUIPMENT																
COMMAND AND CONTROL					X	X						X				X
COMMUNICATIONS										X	X			X	X	
INFORMATION MANAGEMENT		X														
NAVIGATION	X			X												
OPERATIONAL FLIGHT PROGRAMS			X					X								
RANGE OPERATIONS																
SIMULATOR AND TRAINER																
SUPPORT									X				X			
SURVEILLANCE AND WARNING							X									

	49 SK SATELLITE CONTROL	50 STEM	51 TACC AUTO	52 TACS/TADS	53 TIPI-DC/SR	54 TIPI-IAC	55 TIPI-II	56 TIPI-TERPE	57 TOSS	58 TRACALS-PIDP	59 TRACALS-VFR	60 TRI-TAC TCCF	61 USAF TFWC SUPPORT	62 WILD WEASEL	63 WWMCCS	64 WWMCCS II
AUTOMATIC TEST EQUIPMENT																
COMMAND AND CONTROL	X		X	X	X	X	X	X							X	
COMMUNICATIONS									X			X				
INFORMATION MANAGEMENT																
NAVIGATION										X						
OPERATIONAL FLIGHT PROGRAMS														X		
RANGE OPERATIONS													X			
SIMULATOR AND TRAINER		X									X					
SUPPORT																X
SURVEILLANCE AND WARNING																

Table II

Reporting Commands/Organizations by System

		ATE		CC																
		ACTS	ATEC	AFSCF	COMBAT GRANDE	CCPDS	DMSP GROUND SEGMENT	E-3A AWACS	NORAD CMC	PACOM C^4	SACCS/FMIS	SDS	SK SATELLITE CONTROL	TACC AUTO	TACS/TADS	TIPI-DC/SR	TIPI-IAC	TIPI-II	TIPI-TERPE	WWMCCS
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	ESD		X		X			X	X					X		X	X	X	X	X
	MITRE		X		X			X	X	X				X						X
	SAC/ADXRM					X					X									
	SAC/DOKS																			
	SAMSO	X		X			X					X	X							
	TAC/ADY														X	X				
	MAC																			
	AFAL																			
	ADTC																			
	ASD																			
STATUS	PLANNING																		X	
	SOURCE SELECTION																			
	DEVELOPMENT/ TESTING				X			X	X					X		X	X			
	OPERATIONAL	X		X		X	X			X	X	X			X			X		
	MIXTURE		X										X							X

		COMM										IM				NAV		
		AFSATCOM I	AFSATCOM II/III	E-4 BLOCK I	JTIDS/ASIT	RTF	SACCS DTS	SATIN I	SATIN IV	TRI-TAC TCCF	TOSS	AFEES	E-4 BLOCK II	IDHS	MACIMS	LORAN c/d GROUND CHAIN	NAVSTAR GPS	TRACALS-PIDP
		20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
	ESD	X	X	X	X				X	X		X	X		X	X		X
	MITRE	X	X	X					X	X			X		X			
	SAC/ADXRM					X					X			X				
	SAC/DOKS						X	X										
	SAMSO																X	
	TAC/ADY																	
	MAC														X			
	AFAL																	
	ADTC																	
	ASD																	
STATUS	PLANNING												X					
	SOURCE SELECTION				X				X									X
	DEVELOPMENT/ TESTING	X		X						X	X					X	X	
	OPERATIONAL					X	X	X				X		X				
	MIXTURE		X												X			

		OFP											RO				
		AFAL	B-1 BOMBER	C-5 CARGO	DMSP SPACE SEGMENT	EF-111A	F-15	F-16	LORAN AN/ARN-101	MINUTEMAN III	PELSS	WILD WEASEL	ADTC/TSX	ASROS	CBDRO	GERTS	USAF TFWC SUPPORT
		37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52
	ESD								X								
	MITRE																
	SAC/ADXRM																
	SAC/DOKS																
	SAMSO				X					X				X	X	X	
	TAC/ADY																X
	MAC																
	AFAL	X															
	ADTC												X				
	ASD		X	X		X	X	X			X	X					
STATUS	PLANNING										X						
	SOURCE SELECTION																
	DEVELOPMENT/ TESTING	X	X			X		X	X					X			
	OPERATIONAL			X	X		X			X		X	X		X	X	X
	MIXTURE																

		ST		SUP					SW					
		STEM	TRACALS-VFR	DMSP COMMAND SUPPORT	DS&A	RISS	SACOPS	WWMCCS II	COBRA DANE	CONUS OTH	GEODSS	JSS	PAVE PAWS	
		53	54	55	56	57	58	59	60	61	62	63	64	
	ESD	X	X						X	X	X	X	X	29
	MITRE	X								X	X	X	X	19
	SAC/ADXRM			X		X	X	X						9
	SAC/DOKS													2
	SAMSO				X									12
	TAC/ADY													3
	MAC													1
	AFAL													1
	ADTC													1
	ASD													7
STATUS	PLANNING										X	X		5
	SOURCE SELECTION	X												4
	DEVELOPMENT/ TESTING		X						X	X			X	22
	OPERATIONAL			X	X	X	X	X						28
	MIXTURE													5

Table III

Systems-Hardware

LEGEND

■ INDICATES MAIN COMPUTER

D INDICATES DEVELOPMENT COMPUTER

	ACTS (1)	ATEC (2)	AFSCF (3)	COMBAT GRANDE (4)	CCPDS (5)	DMSP GROUND SEGMENT (6)	E-3A AWACS (7)	NORAD CMC (8)	PACOM C^4 (9)	SACCS/FMIS (10)	SDS (11)	SK SATELLITE CONTROL (12)	TACC AUTO (13)	TACS/TADS (14)	TIPI-DC/SR (15)	TIPI-IAC (16)	TIPI-II (17)	TIPI TERPE (18)	WWMCCS (19)
	ATE		CC																
BURROUGHS 700/D MACHINE																			
CDC 5600 (1700), AN/UYK-25													X						
CDC 6000/7000 SERIES, CYBER 70 SERIES																			
DEC SYSTEM 10																			
HONEYWELL 6000 SERIES								■	■	■									■
HUGHES 118 SERIES				■										■					
IBM 360/370							D										D		
IBM 7090 SERIES						X													
UNIVAC 1100 SERIES, AN/UYK-7					■	X		X					■		■	■	■	■	
UNIVAC SERIES 70																			
UNIVAC 1600 SERIES, AN/UYK-20, AN/UYK-15																■			
CDC 3000 SERIES			X								■	■							
DATA GENERAL NOVA, ROLM 1600, AN/UYK-12						■		■							■				
DEC PDP-8																			
DEC PDP-11				X															
DEC PDP-15																			
HARRIS S-120																			
HONEYWELL 16 SERIES DATANET 355		X						X	X										
IBM SYSTEM/4P1							■												
MODCOMP																			
RAYTHEON RDS-500																			
XEROX 550, SIGMA SERIES																			
OTHER (MICRO & FLT COMPUTERS)	X		X	X													X		
UNKNOWN																			

	AFSATCOM I (20)	AFSATCOM II/III (21)	E-4 BLOCK I (22)	JTIDS/ASIT (23)	RTF (24)	SACCS DTS (25)	SATIN I (26)	SATIN IV (27)	TRI-TAC TCCF (28)	TOSS (29)	AFEES (30)	E-4 BLOCK II (31)	IDHS (32)	MACIMS (33)	LORAN c/d GROUND CHAIN (34)	NAVSTAR GPS (35)	TRACALS-PIDP (36)
	COMM										DM				NAV		
BURROUGHS 700/D MACHINE			X														
CDC 5600 (1700), AN/UYK-25																	
CDC 6000/7000 SERIES, CYBER 70 SERIES																	
DEC SYSTEM 10										■							
HONEYWELL 6000 SERIES													■	■			
HUGHES 118 SERIES																	
IBM 360/370		■											■				
IBM 7090 SERIES																	
UNIVAC 1100 SERIES, AN/UYK-7	D																
UNIVAC SERIES 70							■										
UNIVAC 1600 SERIES, AN/UYK-20, AN/UYK-15							■		■						■		
CDC 3000 SERIES																X	
DATA GENERAL NOVA, ROLM 1600, AN/UYK-12	■				■												
DEC PDP-8																	
DEC PDP-11											■						
DEC PDP-15																	
HARRIS S-120																	
HONEYWELL 16 SERIES DATANET 355														X			
IBM SYSTEM/4P1																	
MODCOMP																	
RAYTHEON RDS-500																	
XEROX 550, SIGMA SERIES																X	
OTHER (MICRO & FLT COMPUTERS)						X										X	
UNKNOWN				X				X				X					X

	AFAL (37)	B-1 BOMBER (38)	C-5 CARGO (39)	DMSP SPACE SEGMENT (40)	EF-111A (41)	F-15 (42)	F-16 (43)	LORAN AN/ARN-101 (44)	MINUTEMAN III (45)	PELSS (46)	WILD WEASEL (47)	ADTC/TSX (48)	ASTROS (49)	CSDRO (50)	GERTS (51)	USAF TFWC SUPPORT (52)	STEM (53)	TRACALS-VFR (54)	DMSP COMMAND SUPPORT (55)	D&A (56)	RISS (57)	SACOPS (58)	WWMCCS II (59)	COBRA DANE (60)	CONUS OTH (61)	GEODSS (62)	JSS (63)	PAVE PAWS (64)
	OFP											RO				ST		SUP					SW					
BURROUGHS 700/D MACHINE																												
CDC 5600 (1700), AN/UYK-25																												
CDC 6000/7000 SERIES, CYBER 70 SERIES				D								X				■				■				■				■
DEC SYSTEM 10	X																											
HONEYWELL 6000 SERIES																			■			■	■					
HUGHES 118 SERIES																												
IBM 360/370							D	D	D			X	X	X						■		■						
IBM 7090 SERIES															■													
UNIVAC 1100 SERIES, AN/UYK-7										D															X			
UNIVAC SERIES 70																												
UNIVAC 1600 SERIES, AN/UYK-20, AN/UYK-15										D															X			
CDC 3000 SERIES																					■			X				
DATA GENERAL NOVA, ROLM 1600, AN/UYK-12																										■		
DEC PDP-8																X												
DEC PDP-11												X												X				
DEC PDP-15												X				X												
HARRIS S-120															X													
HONEYWELL 16 SERIES DATANET 355																												
IBM SYSTEM/4P1						■																						
MODCOMP																												■
RAYTHEON RDS-500																								X				
XEROX 550, SIGMA SERIES																■												
OTHER (MICRO & FLT COMPUTERS)		X	X	X	X	X	X	X	X	X	X					X		X			X			X				X
UNKNOWN					X						X						X										X	

Table IV

Systems-Language

LEGEND

( ) INDICATES INCIDENTAL USE

? INDICATES POSSIBLE USE

	ATE		CC																
	ACTS 1	ATEC 2	AFSCF 3	COMBAT GRANDE 4	CCPDS 5	DMSP GROUND SEGMENT 6	E-3A AWACS 7	NORAD CMC 8	PACOM C4 9	SACC/FMIS 10	SDS 11	SK SATELLITE CONTROL 12	TACC AUTO 13	TACS/TADS 14	TIPI-DC/SR 15	TIPI-IAC 16	TIPI-II 17	TIPI-TERPE 18	WWMCCS 19
ASSEMBLY	X	X		(X)		(X)	(X)	(X)					X	X	(X)	(X)	X	X	X
ALGOL																			(X)
APL																			
ATLAS						X													
BASIC	X																		(X)
CMS-2																X			
COBOL								(X)	X										X
DSPL																			
FORTRAN IV		(X)				X	(X)	X	(X)										X
FORTRAN V [1]						X													
JOVIAL(J3)				X	X		X	X		X			X		X	X		X	X
JOVIAL(J3B)																			
JOVIAL(J4)			X								X	X							
JOVIAL J73/I																			
MUMPS II																			
SIMSCIPT					X														(X)
SPL																			
HOL																			
HOL and/or ASSEMBLY																			

	COMM										IM				NAV		
	AFSATCOM I 20	AFSATCOM II/III 21	R-4 BLOCK I 22	JTIDS/ASIT 23	RTF 24	SACCS DTS 25	SATIN I 26	SATIN IV 27	TRI-TAC TCCF 28	TOSS 29	AFEES 30	E-4 BLOCK II 31	IDHS 32	MACIMS 33	LORAN c/d GROUND CHAIN 34	NAVSTAR GPS 35	TRACALS-PIDP 36
ASSEMBLY	X		X	?	X	X	X	?	X	X		?			X	X	X
ALGOL																	
APL		X															
ATLAS																	
BASIC									X								
CMS-2				?													
COBOL							(X)						X	X			
DSPL									X								
FORTRAN IV		X		?			(X)		X				X			X[2]	
FORTRAN V [1]																	
JOVIAL(J3)				?								X					
JOVIAL(J3B)																	
JOVIAL(J4)																X	
JOVIAL J73/I																	
MUMPS II											X						
SIMSCIPT																	
SPL																	
HOL				X				X									
HOL and/or ASSEMBLY																	

	OFP											RO			
	AFAL 37	B-1 BOMBER 38	C-5 CARGO 39	DMSP SPACE SEGMENT 40	EF-111A 41	F-15 42	F-16 43	LORAN AN/ARN-101 44	MINUTEMAN III 45	PELSS 46	WILD WEASEL 47	ADTC 48	ASTROF 49	CSDRO 50	GERTS 51
ASSEMBLY	(X)	(X)	X	X	X	X	X	X	(X)	(X)	X	(X)	X	(X)	
ALGOL															
APL															
ATLAS															
BASIC															
CMS-2															
COBOL														X	
DSPL															
FORTRAN IV									X			X	X[3]	X	X
FORTRAN V [1]															
JOVIAL(J3)										?					
JOVIAL(J3B)	X	X					X			?					
JOVIAL(J4)															
JOVIAL J73/I	X									?		X			
MUMPS II															
SIMSCIPT															
SPL										X					
HOL															
HOL and/or ASSEMBLY				X											

	RO	ST		SUP					SW				
	USAF TFWC SUPPORT 52	STEM 53	TRACALS-VFR 54	DMSP COMMAND SUPPORT 55	DS&A 56	RISS 57	SACOPS 58	WWMCCS II 59	COBRA DANE 60	CONUS OTH 61	GEODSS 62	JSS 63	PAVE PAWS 64
ASSEMBLY	(X)	(X)		X		(X)	X		(X)	X	X	(X)	(X)
ALGOL													
APL	(X)												
ATLAS													
BASIC	(X)												
CMS-2													
COBOL	(X)						X	X					
DSPL													
FORTRAN IV	X	?	X	X	X	X	X	X	X				X
FORTRAN V [1]										(X)			
JOVIAL(J3)		?						X		X			X
JOVIAL(J3B)													
JOVIAL(J4)													
JOVIAL J73/I													
MUMPS II													
SIMSCIPT													
SPL		X										X	
HOL													
HOL and/or ASSEMBLY													

[1] SPECIAL UNIVAC FORTRAN V   [2] USES MELTRAN PREPROCESSOR   [3] USES S-FORTRAN PREPROCESSOR

Table V

Compilers

Legend
( ) indicates compiler that is used in reported system

	Burroughs 700/D Machine	CDC 5600 (1700), AN/UYK-25	CDC 6000/7000, CYBER 70 Series	DEC System 10	Honeywell 6000 Series*	Hughes 118 Series	IBM 360/370	IBM 7090 Series	UNIVAC 1100 Series, AN/UYK-7	UNIVAC Series 70	UNIVAC 1600, AN/UYK-20, AN/UYK-15	CDC 3000 Series	Data General Nova, Rolm 1601, AN/UYK-12	DEC PDP-8	DEC PDP-11	DEC PDP-15	Harris S-120	Honeywell 16 Series, Datanet 355	IBM System/4P1	MODCOMP	RAYTHEON RDS-500	Xerox 550, Sigma Series
ALGOL	X		X	X	X		X		X			X	X	X	X	X	X					
BASIC		X	X	X	X		X		X		(X)	X	X	X	X		X	X		X		X
COBOL	X		X	X	(X)		(X)		X	(X)	X	X			X		X					X
CMS-2											(X)											
DSPL											(X)											
FORTRAN		X	X	X	(X)		(X)	(X)	X	(X)	(X)	X	(X)	(X)	(X)	(X)	(X)	(X)		(X)	(X)	X
JOVIAL (J3)			(X)		(X)	(X)	(X)		(X)													
JOVIAL (J3B)				(X)																		
JOVIAL (J4)												(X)										
JOVIAL J73/I				(X)																		
MUMPS II															(X)							
SIMSCRIPT					(X)				(X)													
APL				X			(X)		X													X
SPL			(X)																			
PL/I			X				X		X													

( ) indicates compiler that is used in reported system
* under GCOS

Table VI

Criteria Affecting Language Selection

	1 ACTS	2 ATEC	3 AFSCF	4 COMBAT GRANDE	5 CCPDS	6 DMSP GROUND SEGMENT	7 E-3A AWACS	8 NORAD CMC	9 PACOM C4	10 SACCS/FMIS	11 SDS	12 SK SATELLITE CONTROL	13 TACC AUTO	14 TACS/TADS	15 TIPI-DC/SR	16 TIPI-IAC	17 TIPI-II	18 TIPI-TERPE	19 WWMCCS
	ATE		CC																
REQUIREMENT: AIR FORCE					X		X	X					X						
REQUIREMENT: USER/SPO			X			X				X	X	X			X	X		X	X
REQUIREMENT: USER/SPO (CLASS)[1]				X															
DEVELOPER DISCRETION: CONTRACTOR	X	X		X													X		
DEVELOPER DISCRETION: USER									X										
DEVELOPER DISCRETION: AF - OTHER ORG.																			
OVERALL SYSTEM DESIGN																			
SUITABILITY FOR APPLICATION			X	X		X	X		X			X			X	X		X	
PROCESSING REQUIREMENTS		X																	
HARDWARE SELECTION		X		X		X				X	X		X				X		X
OFF-THE-SHELF APPROACH													X						
AVAILABILITY OF COMPILERS				X		X													
PROGRAMMER TRAINING	X				X	X			X										
MAINTAINABILITY/ RELIABILITY			X																
SOFTWARE TRANSPORTABILITY					X			X	X	X								X	
USER EXPERIENCE							X	X	X							X		X	X
STANDARD LANGUAGE																			
REUSE OF COMPILERS											X	X				X		X	X
SOFTWARE ENGINEERING SUPPORT																			
UNKNOWN														X					

	20 AFSATCOM I	21 AFSATCOM II/III	22 E-4 BLOCK I	23 JTIDS/ASTT	24 RTF	25 SACCS DTS	26 SATIN I	27 SATIN IV	28 TRI-TAC TCCF	29 TOSS	30 AFEES	31 E-4 BLOCK II	32 IDHS	33 MACIMS
	COMM										IM			
REQUIREMENT: AIR FORCE												X		X
REQUIREMENT: USER/SPO									X					
REQUIREMENT: USER/SPO (CLASS)[1]				X										
DEVELOPER DISCRETION: CONTRACTOR	X		X	X	X	X	X	X		X	X			
DEVELOPER DISCRETION: USER														
DEVELOPER DISCRETION: AF - OTHER ORG.		X												
OVERALL SYSTEM DESIGN				X				X						
SUITABILITY FOR APPLICATION		X			X		X		X		X			X
PROCESSING REQUIREMENTS					X	X	X							
HARDWARE SELECTION	X	X	X		X				X					
OFF-THE-SHELF APPROACH				X				X						
AVAILABILITY OF COMPILERS		X												
PROGRAMMER TRAINING												X		X
MAINTAINABILITY/ RELIABILITY														
SOFTWARE TRANSPORTABILITY									X					
USER EXPERIENCE												X		X
STANDARD LANGUAGE														X
REUSE OF COMPILERS														
SOFTWARE ENGINEERING SUPPORT														
UNKNOWN													X	

	34 LORAN c/d GROUND CHAIN	35 NAVSTAR GPS	36 TRACALS PIDP	37 AFAL	38 B-1 BOMBER	39 C-5 CARGO	40 DMSP SPACE SEGMENT	41 EF-111A	42 F-15	43 F-16	44 LORAN AN/ARN-101	45 MINUTEMAN III	46 PELSS	47 WILD WEASEL
	NAV			OFP										
REQUIREMENT: AIR FORCE														
REQUIREMENT: USER/SPO		X										X[4]		
REQUIREMENT: USER/SPO (CLASS)[1]					X					X			X	
DEVELOPER DISCRETION: CONTRACTOR	X		X		X	X	X	X	X	X	X	X[4]	X	X
DEVELOPER DISCRETION: USER														
DEVELOPER DISCRETION: AF - OTHER ORG.				X										
OVERALL SYSTEM DESIGN											X			
SUITABILITY FOR APPLICATION		X	X	X	X		X			X		X	X	X
PROCESSING REQUIREMENTS			X	X	X			X			X			
HARDWARE SELECTION						X	X	X	X	X		X		X
OFF-THE-SHELF APPROACH	X		X							X[5]				
AVAILABILITY OF COMPILERS		X												
PROGRAMMER TRAINING		X			X					X		X		
MAINTAINABILITY/ RELIABILITY		X			X					X			X	
SOFTWARE TRANSPORTABILITY		X												
USER EXPERIENCE														
STANDARD LANGUAGE												X		
REUSE OF COMPILERS		X												
SOFTWARE ENGINEERING SUPPORT		X												
UNKNOWN														

	48 ADTC/TSX	49 ASTROS	50 CSDRO	51 GERTS	52 USAF TFWC SUPPORT	53 STEM	54 TRACALS VFR	55 DMSP COMMAND SUPPORT	56 DS&A	57 RISS	58 SACOPS	59 WWMCCS II	60 COBRA DANE	61 CONUS OTH	62 GEODSS	63 JSS	64 PAVE PAWS
	RO					ST		SUP					SW				
REQUIREMENT: AIR FORCE														X		X[2]	X
REQUIREMENT: USER/SPO	X				X			X	X	X							
REQUIREMENT: USER/SPO (CLASS)[1]				X		X							X			X	
DEVELOPER DISCRETION: CONTRACTOR				X		X	X						X			X	
DEVELOPER DISCRETION: USER		X	X														
DEVELOPER DISCRETION: AF - OTHER ORG.															X		
OVERALL SYSTEM DESIGN																X	
SUITABILITY FOR APPLICATION	X	X	X	X	X			X	X	X			X				
PROCESSING REQUIREMENTS															X		
HARDWARE SELECTION													X		X		
OFF-THE-SHELF APPROACH						X										X	
AVAILABILITY OF COMPILERS							X		X				X			X	X[3]
PROGRAMMER TRAINING	X	X	X			X				X			X		X		
MAINTAINABILITY/ RELIABILITY		X				X											X
SOFTWARE TRANSPORTABILITY				X	X			X	X								
USER EXPERIENCE			X											X			
STANDARD LANGUAGE			X														
REUSE OF COMPILERS																	
SOFTWARE ENGINEERING SUPPORT		X															
UNKNOWN											X	X					

[1] CLASS OF LANGUAGE [2] WAIVER REQUESTED [3] FORTRAN AVAILABLE ON MINIS [4] DECISION-MAKER DIFFERENT FOR EACH OF TWO COMPILERS

[5] J3B AND J3B-1, NOT J3B-2

Table VII

Software Development and Maintenance Responsibility

LEGEND

X - SYSTEM/SUPPORT SOFTWARE

O - APPLICATION SOFTWARE

▣ - BOTH

			DEVELOPMENT: CONTRACTOR	DEVELOPMENT: USER	DEVELOPMENT: OTHER AF ORG.	DEVELOPMENT: UNKNOWN	MAINTENANCE: CONTRACTOR	MAINTENANCE: USER	MAINTENANCE: OTHER AF ORG.	MAINTENANCE: UNKNOWN	AF AGENCY - WWMCCS
ATE	1	ACTS	▣							▣	
	2	ATEC	▣						▣		
CC	3	AFSCF	▣				▣				
	4	COMBAT GRANDE	▣					X			
	5	CCPDS	X	O						▣	
	6	DMSP GROUND SEGMENT	▣							▣	
	7	E-3A AWACS	▣					▣			
	8	NORAD CMC	X	O				O	X		X
	9	PACOM C^4	X	O				O	X		X
	10	SACCS/FMIS	X	O				O	X		X
	11	SDS	O		[2] X		[2] ▣				
	12	SK SATELLITE CONTROL	O		[2] X		[2] ▣				
	13	TACC AUTO	▣	O				▣			
	14	TACS/TADS				▣				▣	
	15	TIPI-DC/SR	▣					▣			
	16	TIPI-IAC			[3] X						X
	17	TIPI-II	▣					▣			
	18	TIPI-TERPE	▣					▣			
	19	WWMCCS	X	O					▣		▣
COMM	20	AFSATCOM I	▣						▣		
	21	AFSATCOM II/III	X	O						▣	
	22	E-4 BLOCK I	X	O						▣	
	23	JTIDS/ASIT	▣							▣	
	24	RTF	▣							▣	
	25	SACCS DTS	X	O					▣		
	26	SATIN I	X	O				O		X	
	27	SATIN IV	X		O				▣		
	28	TRI-TAC TCCF	X	O					▣		
	29	TOSS	X			O				▣	
	30	AFEES	▣							▣	
DM	31	E-4 BLOCK II	X	O				O			X
	32	IDHS	X			O				▣	
	33	MACIMS	X	O				O	X		X
NAV	34	LORAN c/d GROUND CHAIN	▣						[1] ▣		
	35	NAVSTAR GPS	X	O			[2] ▣				
	36	TRACALS PIDP	▣						▣		
OFP	37	AFAL	▣							▣	
	38	B-1 BOMBER	X			O				▣	
	39	C-5 CARGO	▣				▣				
	40	DMSP SPACE SEGMENT	▣							▣	
	41	EF-111A	▣							▣	
	42	F-15	X		O		▣				
	43	F-16	▣				▣				
	44	LORAN AN/ARN-101	▣					▣			
	45	MINUTEMAN III	▣							▣	
	46	PELSS	▣						▣		
	47	WILD WEASEL	▣				▣				
RO	48	ADTC/TSX	▣							▣	
	49	ASTROS	X	O			▣				
	50	CSDRO	X	O			▣				
	51	GERTS	▣							▣	
	52	USAF TFWC SUPPORT	X	O			X	O			
ST	53	STEM	▣				▣				
	54	TRACALS-VFR	▣						▣		
SUP	55	DMSP COMMAND SUPPORT	X	O				O	X		X
	56	DS&A	▣				[2] ▣				
	57	RISS	X	O				O		X	
	58	SACOPS	X	O			X	O			
	59	WWMCCS II	X	O				O	X		X
SW	60	COBRA DANE	▣				▣				
	61	CONUS OTH	▣							▣	
	62	GEODSS	X		O					▣	
	63	JSS	▣							▣	
	64	PAVE PAWS	▣						▣		

[1] CURRENTLY MAINTAINED BY UNIVAC [2] USES AFSCF [3] NAVAL SURFACE WEAPONS CENTER

SECTION III

AIR FORCE EXPERIENCE BY APPLICATION AREA

Systems in the application areas described in Section II reflect different hardware/software environment concerns, computer programming use, language decision-making factors, language standard adherence, and software development and maintenance dependencies. Based on the data tabulated in Section II and the material reported during Phase I of the HOL Standardization Program [LAPA76], the patterns which emerge within each application area are presented in this section.

AUTOMATIC TEST EQUIPMENT

Only two Automatic Test Equipment (ATE) systems were reported, one by SAMSO which is operational and one by ESD and MITRE for which Phase I is in development and Phase II is being planned.

Hardware/Software Environment

Both systems used commercially available minicomputers, operating systems, compilers, and support software; distributed microcomputers are being considered for Phase II of ESD's ATEC.

Languages Used

Assembly language was used in both systems; HP BASIC was used in one reported system.

In one system assembly language was used for all test functions, especially terminal test drivers and bit error-rate testing. It also supported near real-time requirements and data base management.

In the other system HPBASIC was used for all test functions, I/O utilities, and configuration utilities.

Language Selection

In both cases the contractor selected the language based on his experience, hardware selected, processing requirements, and his programmers' backgrounds.

Language Standards

No language standards were required or used.

Software Development and Maintenance

Both systems were developed by contractors, both experienced some difficulty with managing software development. The Air Force will maintain both.

Unlike ASD systems, no central general-purpose computer was used for program development.

Relation to Phase I

ASD ATE systems, i.e., ATE for avionics systems, and their software-related concerns are described in [LAPA76] as part of the HOL Standardization Program Phase I results.

ASD systems represent slightly different modes of development and use; they use minicomputers for equipment testing but use large-scale computers for compiling and maintaining production programs.

ATLAS is used frequently by ASD ATE systems. Until a recent DoD decision naming ATLAS and OPAL as the only two allowable HOLs for ATE systems, no standards had applied in this area. Implementation of this recent decision could lead to more uniformity of language use in the future. Use of ATLAS, [ARIN75] plus Air Force extensions, as a standard language was a formal recommendation of Phase I of the AFSC HOL Standardization Program; this position was not overturned by data collected from the two new systems in Phase II.

COMMAND AND CONTROL

Seventeen Command and Control (CC) systems are summarized in Volume II. Nine were reported by ESD, six by MITRE, two by SAC/ADXRM, four by SAMSO, and three by TAC/ADY; several systems were described by more than one agency. Eight of these systems are operational, six are in development, one is being planned, and two have segments in more than one stage of the acquisition process.

Hardware/Software Environment

All systems use at least one large-scale ground-based computer. Minicomputers are used as communications, peripheral, or subsystem controllers. Two systems use militarized flight computers, programs

for which are developed on a ground-based computer. Most hardware is commercially available, e.g., H6000 for WWMCCS systems, but tactical systems use militarized equipment, e.g., AN/UYK-7 for TACC AUTO.

Operating systems and support software that accompany commercially available equipment are used if possible, sometimes with modifications and tailored support packages, such as a data management facility. Several systems have required development of unique operating systems, especially on militarized machines, and/or executives, especially on secondary processors.

Commercially available compilers are used when available with the hardware, primarily FORTRAN compilers, or required by the RFP (Request for Proposal). New compiler development has been required for JOVIAL (J3), JOVIAL (J4), and SPL where compilers were unavailable. WWMCCS hardware and software are used in four reported systems.

Languages Used

The following languages are used by CC systems:

- assembly - 11 including 5 incidental use
- JOVIAL (J3) - 10
- JOVIAL (J4) - 3
- FORTRAN - 5 including 2 incidental use
- COBOL - 3 including 1 incidental use
- ALGOL, ATLAS, CMS-2, SIMSCRIPT - all have limited use

All systems except two (TIPI-II and TACS/TADS) have major HOL use. JOVIAL (J3) is by far the most widely used HOL for CC systems. It is considered "suitable" for performing command and control functions because of features like COMPOOL for defining shared data, block structure, and available data types. J3 is used primarily on large-scale computers; compilers exist for five different commercial computer lines.

JOVIAL (J3) compilers are commercially available on UNIVAC 1108, Honeywell 6000, and CDC Cyber machines. JOVIAL compilers for other processors were developed specifically for the computer under contract to the Air Force; the JOVIAL Compiler Implementation Tool (JOCIT) has been used in one case (WWMCCS) to reduce the cost of

compiler development. AF-owned J3 compilers exist for Honeywell 6000 (JOCIT JOVIAL), Hughes 118, IBM 360/370, and AN/UYK-7 (two different compilers) machines; in addition cross-compilers from IBM 360/370 to IBM 4Pi/CC-1 and AN/UYK-7 to AN/UYK-25 are in use.

Assembly language is used to perform real-time functions where object code efficiency is important, for example to keep up with radar inputs. It is used to supplement HOLs on large-scale computers and to program minicomputers without suitable HOLs (n.b. JOVIAL is available only for the AN/UYK-25 and 4Pi CC-1).

FORTRAN is used for scientific computations, a supporting function in command and control systems. It is used primarily on minicomputers, e.g., in Defense Meteorological Space Program (see Volume II), and radar or navigation computers, e.g., in E-3A.

JOVIAL (J4) is used exclusively at the Air Force Satellite Control Facility's Satellite Test Center. It is a unique version of JOVIAL implemented on the CDC 3800; J4 is similar to J3, but with additional I/O facilities.

COBOL is used in applications with heavy data processing components, e.g., at PACOM, to perform file maintenance, data retrieval and formatting, and report production functions.

Language Selection

Languages for command and control systems usually (i.e., for 13 of the 17 systems) are selected by:

- Air Force requirement (4 systems). Since AFR 300-10 [AIRF71] requires use of JOVIAL for CC applications, FORTRAN for scientific applications, and COBOL for data processing, Air Force requirement is seen as the primary reason for selection of these languages.

- User or SPO requirement placed on the developer for a specific language (9 systems). These systems were either not perceived to be within the jurisdiction of AFR 300-10 or user needs were perceived to be a stronger influence on the language decision than the formal requirement, e.g., SAC's experience with JOVIAL dictates continued use of JOVIAL. All but one of these systems are using a version of JOVIAL, either J3 or J4, as the principal language.

Factors which influenced the language selection decision are:

- suitability for the application - (9 systems, primarily JOVIAL)
- hardware selection - (6 systems)
- software transportability - primarily of existing FORTRAN or JOVIAL programs to new systems (5 systems)
- user experience - (6 systems)
- reuse of compiler - J4 compiler and WWMCCS compilers (5 systems)
- programmer training - (5 systems)
- availability of compilers - (2 systems)
- off-the-shelf approach - (1 system)
- maintainability/reliability - (1 system)

Language Standards

AFR 300-10 [AIRF71] requires that JOVIAL (J3) be used for command and control applications; this requirement was cited by four systems. AFM 100-24 [AIRF61] defines the J3 language; almost all compilers deviate somewhat from or exceed the standard. At least two compilers (for E-3A and TACC AUTO) were tested using the JOVIAL Compiler Validation System (JCVS) [FELT76].

AFR 300-10 also requires the use of FORTRAN, as defined by ANS X3.9-1966 [ANSI66A] or X3.10-1966 [ANSI66B], for advanced mathematical applications and COBOL, as defined by ANSI X3.23-1968 [ANSI68], for data processing applications. All compilers reported generally implement extensions to the relevant standard.

WWMCCS (World-Wide Military Command and Control System) is a family of systems, some of which are the Air Force's responsibility (see Volume II for details). AFM 171-100 [AIRF74] contains Air Force Automated Data Systems (ADS) Standards. Volume I includes language standards for systems developed by AFDSDC; Volume II includes HIS 6000 and WWMCCS standards; and Volume III includes base level data processing or B3500 standards. Language standards reference AFR 300-10 and itemize specific language feature requirements for COBOL.

Adherence to the COBOL, FORTRAN, and JOVIAL standards itemized in AFR 300-10 was required with the initial WWMCCS purchase. JOCIT JOVIAL later replaced the initial JOVIAL compiler. Currently compilers for each language (despite minor deviations from the Air Force standards), certain application software packages, support software, and the operating system have been standardized for all Air Force WWMCCS systems and future acquisitions of similar hardware (Honeywell 6000s). Mechanisms for establishing new WWMCCS-standard software and for maintaining existing software have been established; emphasis is on reuse of software leading to de facto standardization.

Software Development and Maintenance

For ESD-reported systems, application software is generally written by a contractor or subcontractor. Sometimes, e.g., for tactical systems, Air Force personnel are assigned to assist in the software development which builds in-house expertise. After system delivery, the user, e.g., the Major Command, performs maintenance at a central system support facility.

Application software for SAC command and control systems is developed and maintained by Air Force personnel.

Application software for SAMSO-reported systems which use the Air Force Satellite Control Facility (AFSCF) is developed and maintained by contractors; AFSCF has no plans for an organic computer programming capability.

CCTC (Command and Control Technology Center) performs operating system and support software maintenance for WWMCCS systems.

Relation to Phase I

Many of the same systems were surveyed in Phase I of the AFSC HOL Standardization Program [LAPA76], but are covered here in greater depth. The same trends are evident; CC systems lack a specific language/selection methodology but software transportability and reuse of compilers show up more clearly as influences here than in Phase I. Research on new languages for CC applications was covered in [LAPA76] and not repeated here. CC remains a multiple HOL environment with JOVIAL (J3) the most widely used HOL.

COMMUNICATIONS

Ten Communications (COMM) systems are summarized in Volume II. Two systems are transmission media, three are terminals, three are message switches, and two are networks. ESD reported on six systems, MITRE on five, SAC/ADXRM on two, and SAC/DOKS on three; several systems were described by more than one agency. Four of these systems are in development or testing, three are operational, two are in source selection, and one is being planned.

Hardware/Software Environment

Hardware for COMM systems is usually in the medium to small computer range. Half the systems require militarized hardware while the other half use commercially available equipment. In new acquisitions, message handling multi-computer networks appear to be replacing single processor switches.

Most systems have unique executives or modified versions of commercially provided operating systems; all of these are written in assembly language. Most system and support software is machine dependent since it is written in assembly language.

Languages Used

The following languages are used by COMM systems:

- assembly - 7 plus 2 possible
- FORTRAN - 3 including 1 incidental use
- DSPL - 1
- BASIC - 1
- APL - 1
- HOL and/or assembly - 2; FORTRAN, JOVIAL (J3), or CMS-2 are the only HOLs allowed for JTIDS/ASIT while any language is acceptable for SATIN IV.
- COBOL - incidental use

Assembly language dominates; all seven systems which are past source selection have major assembly language use. Five systems use no other language for application programming; only the MITRE concept development effort, AFSATCOM II/III, uses HOL exclusively for

preliminary calculations. Assembly language is used for all communications functions, especially time-critical ones such as message processing.

FORTRAN is used for batch scientific computations in a few systems. Some existing programs were reused.

TRI-TAC/TCCF uses two interactive HOLs, BASIC and DSPL. DSPL, intended specifically for the on-line development of interactive display programs, was required by the Program Office, defined by the PO in conjunction with MITRE and the contractor, and developed by the contractor, Sperry-UNIVAC.

Communications programming functions have been considered unique and time-critical, leading to dependence on assembly language. Evaluation of existing programming languages for their suitability to COMM processing [SOFT76A] and specification of a new Communications Oriented Language (COL) [BBN76] are underway. No high order language has yet been tried on a U.S. full-scale communications system, although the two systems now in source selection, JTIDS/ASIT and SATIN IV, may change that.

Language Selection

Languages, usually assembly, are selected for most COMM systems at the contractor's discretion. Notable exceptions are TRI-TAC/TCCF's requirement for DSPL, JTIDS/ASIT's requirement for one of three HOLs and the developer's (MITRE) choice of APL for AFSATCOM II/III study.

Factors which heavily influenced the language selection decision are:

- hardware selection - assembler was available on chosen hardware (5 systems)
- suitability for application - (4 systems including 2 HOLs)
- processing requirements - (3 systems)
- overall system design - (2 systems in source selection)
- off-the-shelf approach - (2 systems)

Language Standards

No language standards apply to the COMM area. HOLs are beginning to be used.

Software Development and Maintenance

Contractor develops application and system software for most systems reported. Air Force generally performs maintenance; CCPC (Communications Computer Programming Center) maintains several of the systems reported. CCPC also develops Air Force communications systems, e.g., a portion of SATIN IV (see [LAPA76] for other systems).

Relation to Phase I

By redefining this application area to exclude command and control-related communications functions, patterns focus more clearly than those which appear in [LAPA76] (Phase I). Although not all systems supported by CCPC and reported in Phase I are reported again here, the experience reported earlier with COMM systems is represented. Assembly language use still predominates, while hardware and processing requirements are the driving factors. Experience with HOLs is limited but growing.

INFORMATION MANAGEMENT

Four Information Management (IM) systems are summarized in Volume II. Three were reported by ESD, two by MITRE, one by MAC, and one by SAC/ADXRM; two systems were described by more than one agency. Two systems are operational, one is planned, and one has segments in more than one stage of the acquisition process.

Hardware/Software Environment

IM systems primarily use commercially available hardware, operating systems, compilers, and support software. E-4 Block II is an exception; it plans to use a militarized airborne computer, contractor-developed compiler and cross-compiler, and a ground-based general-purpose computer with architecture similar to the airborne machine for development and maintenance.

WWMCCS hardware and compilers are used for one system; E-4 Block II is part of the WWMCCS family, but WWMCCS hardware and software are not required.

Languages Used

The following languages are used or planned for use by IM systems:

- assembly - 1
- COBOL -2
- FORTRAN - 1
- JOVIAL (J3) - 1
- MUMPS-II - 1

High order languages (HOLs) are used predominantly; no one language emerges as dominant. Each language supports the earlier described IBM functions of information storage, retrieval, display, reporting, and interpretation.

Language Selection

Air Force requirements were cited as the reason for selecting the language in two cases:

- AFR 300-10 [AIRF71] for E-4 Block II, designated a CC system, requires use of JOVIAL (J3).
- AFM 171-100 [AIRF74] dictated use of COBOL in MACIMS.

Contractors decided on the language for AFEES (see Volume II).

Factors which most heavily influenced the language selection decision are:

- suitability for the application - (2 systems)
- programmer training - (2 systems)
- user experience - (2 systems)
- standard language - reflects interest in beginning software development before the system is in-place (1 system)

Language Standards

MACIMS adheres to WWMCCS standards, as required by AFM 171-100 and as discussed under Command and Control, especially in using a COBOL compiler which closely adheres to ANSI X3.23-1968 [ANSI68]. E-4 Block II plans to use JOVIAL (J3) as defined by AFM 100-24.

No other language standards were required or used.

Software Development and Maintenance

SAC and MAC personnel develop and maintain application software. CCTC (Command and Control Technology Center) performs operating system and support software maintenance for WWMCCS systems.

Other software is contractor developed and maintained.

Relation to Phase I

This is a newly identified application area. Two of the four systems are reported here for the first time. The other two, E-4 Block II and MACIMS, were reported under CC in [LAPA76].

NAVIGATION

Three Navigation (NAV) systems are summarized in Volume II. Two systems are in development and one is in source selection.

Hardware/Software Environment

Commercially available minicomputers and associated support software are used in two systems, the militarized AN/UYK-15 and contractor-provided software are used in the LORAN C/D Ground Chain system and unique executive programs and a microprocessor are used in the NAVSTAR GPS system.

The LORAN C/D Ground Chain and TRACALS-PIDP systems both will employ minicomputers and assembly language. NAVSTAR GPS included minicomputers, microprocessors, FORTRAN, JOVIAL (J4), assembly language, and structured programming which are being combined into a highly specialized system. It has three segments (space, control, and user) each of which has equipment and languages tailored to the requirements of the segment.

Languages Used

The following languages are used by NAV systems:

- assembly - all 3
- FORTRAN IV - NAVSTAR only
- JOVIAL (J4) - NAVSTAR only

JOVIAL (J4) is used on the CDC-3800 at the Air Force Satellite Control Facility (AFSCF). FORTRAN is available on the Xerox 550 and HP21MX. In general, assembly language is most heavily used in these NAV systems for performing such functions as real-time process control, display processing, tracking, beacon processing, and inter-facility communications. JOVIAL (J4) is used for satellite control functions; FORTRAN is used for scientific computations, such as for ground tracking and orbit estimation, where there are no severe demands on program size or execution speed.

Language Selection

The principal criterion affecting programming language selection for these NAV systems was contractor choice; JOVIAL (J4) was a requirement of the SPO, partly because of compiler availability.

Other factors are:

- suitability for application - (2 systems)
- off-the-shelf approach - (2 systems)
- processing requirements - (1 system)

Factors peculiar to the use of FORTRAN are:

- programmer training
- maintainability
- software transportability
- software engineering supportability (MELTRAN preprocessor)

Language Standards

No standards were invoked or imposed for the LORAN and TRACALS systems (both of these systems use only assembly language).

In NAVSTAR, standards are:

- minimum use of assembly language
- ANSI Standard ANS 3.9-1966 [ANSI66A] for FORTRAN IV
- AFSCF standard for JOVIAL (J4)

Maintenance

Software maintenance will be performed by the Air Force for two systems, AFLC for LORAN and CCPC for TRACALS. No information on maintenance approach for NAVSTAR was available.

Other Comments

All NAV systems reported are to be delivered by contractors.

MELTRAN (the FORTRAN preprocessor) was chosen for its efficiency of resulting object code. For the NAVSTAR system structured programming is required with emphasis on top-down design, implementation, and testing; the contractor is required to deliver a Computer Programming Manual in response to this requirement.

Relation to Phase I

This is a newly identified application area and all the systems are reported for the first time. No significant patterns emerge.

OPERATIONAL FLIGHT PROGRAMS

Ten Operational Flight Programs (OFP) systems and the experiences of several programs in one laboratory (AFAL) are reported in Volume II. Seven systems were described by ASD, four of which are examples of the traditional assembly language approach to avionics system implementation and three of which are more recent acquisition programs which use high order languages. Four of these avionics (ASD) systems include electronic warfare functions as a portion of overall mission requirements. In addition, two systems were reported by SAMSO and one by ESD. Five of these OFP systems are operational,

four are in development, and one (PELSS) is being planned; AFAL systems are in development.

Hardware/Software Environment

Each system uses one or more militarized airborne computer, most of which are unique to the particular system; 18 different airborne computers are listed in Appendix III. These computers have 16-, 28-, or 32-bit word architectures; some but not all have fixed point hardware.

Each airborne computer has a unique contractor-provided support software package including an assembler. The Air Force has acquired basic support software, such as assemblers and link editors, from contractors. Some attempt has been made to reuse this support software, especially on the EF-111A. Unique, AF-owned compilers and cross-compilers for J3B, J3B-1, J3B-2, J73/I, and FORTRAN have been developed for systems using these HOLs.

Large-scale general-purpose computers, e.g., IBM 360/370 and DEC SYSTEM-10, are used for software development and testing. Commercially available support software plus special simulators, cross-compilers and cross-assemblers are used. Code generated for the airborne computers is loaded via special aerospace ground equipment (AGE).

Languages Used

The following languages are used by OFP systems:

- assembly - all including 3 incidental use
- J3B - 3 versions on 2 ASD systems and one at AFAL
- J73/I - on DAIS and OSC at AFAL
- J3 or J3B or J73/I - on system being planned (PELSS)
- SPL - 1
- FORTRAN IV - 1

Assembly language is used by all systems in varying amounts. It is used primarily for executive functions such as scheduling and interrupt processing, hardware interfacing, input/output control, and other time-critical functions. In addition, space systems use it to

perform command and control, ascent guidance, and telemetry functions. Missile systems use it for missile flight trajectory, command and control interfacing, and command message processing.

Assembly language is used predominantly for electronic warfare functions, but little data is reported in Volume II to indicate why. Data collected during Phase I of the HOL Standardization Program [LAPA76] indicated that Electronic Warfare (EW) functions are highly time-critical and therefore demand efficient object code. This is not corroborated or denied by Phase II data.

JOVIAL (J3B) has been used to perform all avionics functions, except those listed under assembly language. The three existing compilers (J3B, J3B-1, and J3B-2) represent three versions of the language all of which can be compiled by the J3B-2 compiler; each is targeted for a different airborne computer and has unique support software. J3B-1 and J3B-2 support fixed point arithmetic operations but J3B does not.

JOVIAL J73/I is used at AFAL; it replaces J3B on one program (Operational Software Concept) and it is the initial choice on new programs [TRAI76]. All avionics functions, except portions of the executive, have been programmed in J73/I. Comparisons of J73/I and J3B indicate that J73/I produces superior code (see Volume II, AFAL), especially for executive functions.

SPL (Space Programming Language) is used for attitude control software which involves mathematical computations. The SPL compiler was developed via the Compiler Writing System (CWS) at SAMSO.

FORTRAN is used for targeting software and execution-plan data generation which also involve mathematical computations.

<u>Language Selection</u>

Languages for OFP systems are usually selected by contractors either by choosing assembly language when no requirements are levied or by choosing a specific language version when a high order language is required.

Factors which most heavily influenced the language selection decision are:

- suitability to the application - assembly, SPL, JOVIAL, and FORTRAN (8 systems)

- hardware selection - primarily leading to assembly language use (7 systems)

- processing requirements - especially for time-critical functions coded in asssembly language (4 systems)

- programmer training - expected to become easier with use of HOL (3 systems)

- maintainability/reliability - expected to improve with use of HOL (2 systems), improvements achieved (1 system, MINUTEMAN)

- standard language - two compilers were needed since the target machine was not available during development; FORTRAN was chosen to achieve transportability (1 system)

JOVIAL J73/I was initially selected for the B-1 offensive software, but the formal subset had not been defined. At the time no other HOL and compiler were considered suitable for avionics applications [FALK76]. Therefore, an HOL specification was developed under a separate contract based on inputs from the B-1 offensive contractor. The B-1 contractor developed an HOL and compiler for the B-1 offensive software; this became JOVIAL J3B. B-1 defensive software had more severe timing constraints, so fixed point facilities were added to the language resulting in J3B-1. The F-16 contractor produced a new and enhanced J3B compiler (J3B-2).

Language Standards

To date, no official language standards have been imposed on avionics OFP. Softech's documentation of JOVIAL J3B-2 serves to define the language [SOFT76B]. JOVIAL J73/I was defined by a draft specification which was updated to reflect changes required to match the existing implementation; the most recent J73/I specification [RADC76] reflects language improvements.

Space and missile systems have in isolated cases imposed language control. SPL, used in DMSP Space Segment, is defined by a SAMSO Technical Report. FORTRAN, used in MINUTEMAN, was implemented via two compilers and programmers were restricted to using a subset of the language to assure transportability of programs; ANSI X3.9-1966 was not required.

Software Development and Maintenance

OFP software is developed and maintained by contractors. PELSS at ASD is planning for Air Force maintenance. This maintenance is performed at Air Force Logistics Command (AFLC) support sites for some systems, especially those with electronic warfare functions.

Trends in Language Use

OFPs have traditionally depended on assembly language to meet timing and program size requirements outlined earlier in this section (Definition of Application Areas). Use of HOLs is growing, especially to improve reliability and maintainability of software. Experience on the B-1 indicates that recoding software in J3B that was originally coded in assembly language increased space requirements by 20% but took 1/3 the time to code. (Additional discussion of avionics OFP is in [FALK76] and of space and missile system language requirements is in [CALL75].)

Several HOLs suitable to OFP applications are now in use. J73/I has been recommended for avionics programs [FALK76, TRAI76] and any version of JOVIAL is recommended for advanced ballistic missile applications [see MINUTEMAN in Volume II]. Most important is the need to eliminate the development of unique support software for each new system [FALK76, CALL75]; this can be accomplished by increased Air Force control over compilers and support software.

Relation to Phase I

Greater detail on ASD systems is reported here than was available in Phase I [LAPA76]. The details on experiences with J3B and J73/I are reported in [TRAI76] and are not repeated here. Since Phase I, more versions of JOVIAL have been developed and ability of HOLs to perform OFP applications has been shown. The need to reduce proliferation of support software is more apparent now because more systems have had relevant experience, but other issues are not substantially changed.

RANGE OPERATIONS

Three Range Operations (RO) systems and the experiences of two RO agencies, ADTC and SAMTEC, are summarized in Volume II. Two systems are operational and one is in development. SAMTEC is responsible for systems in various stages of acquisition and ADTC is responsible for RO system research and development.

Hardware/Software Environment

All systems use at least one large-scale commercially available computer at a single center; IBM 360/370s and CDC 6000s predominate. Accompanying commercially available operating systems, compilers, and support software are used.

A notable deviation from this practice is the use of S-FORTRAN, a FORTRAN preprocessor, for the ASTROS program at SAMTEC. S-FORTRAN is a commercially supported product which enables FORTRAN programmers to write structured code. Also, under AFAL's direction, ADTC staff is writing the stores management subsystem of DAIS (Digital Avionics Information System) in JOVIAL J73/I via terminal and leased lines.

Languages Used

The following languages are used by RO systems:

- FORTRAN - all
- COBOL - 2
- JOVIAL J73/I - 1
- assembly - incidental use
- APL, BASIC - non-production use

All systems use FORTRAN for scientific calculations including data generation, data reduction, simulation, and other range functions (see Definition of Application Areas). FORTRAN is considered "suitable" to the RO application by the users although extended versions are often used. Subword manipulation is desired to extend the domain of applicability.

COBOL is used for accounting and range scheduling functions which are data processing in nature.

Assembly language is used to handle operating system functions, such as priority input processing and I/O servicing and checking; it supports FORTRAN programs by performing real-time processing and bit extraction.

J73/I is used for a subsystem of DAIS and was required by the lead organization, AFAL.

Language Selection

Languages for Range Operations systems are usually selected by the user:

- In four cases the user required FORTRAN; at SAMTEC FORTRAN is selected for all in-house development

- For GERTS, the user required an HOL and the contractor selected FORTRAN

Factors which most heavily influenced users in making the language selection decision are:

- suitability for the application - all systems

- programmer training - programmers are available with a knowledge of FORTRAN and no training within DoD is required (2 agencies, 1 system)

- software transportability - existing aplication software was transferred to a new system (2 systems)

- standard language, experience of user, maintainability, and software engineering support (S-FORTRAN) all reflect the high degree of availability, commercial support, and dependability of FORTRAN compilers and related support software desired.

Language Standards

The requirement in AFR 300-10 [AIRF71] to use FORTRAN, as defined by ANS X3.9-1966 [ANSI66A] or X3.10-1966 [ANSI66B], for advanced mathematical applications was not cited as a reason for selecting FORTRAN. No system or agency required adherence to the ANS standards, for FORTRAN or COBOL. All used contractor-provided compilers; all compilers implement some extensions to these standards. Systems at USAF TFWC attempted to achieve program transportability by adopting programming standards which restricted programmers to standard FORTRAN features. Programs were not as portable as had been hoped; in the interests of expediency, available extended features and assembly-language subroutines were used, making conversion to the new system a major effort.

Software Development and Maintenance

Range operation software at TFWC is developed and maintained by the Air Force on the primary operational computer. Other RO software, at SAMTEC and for GERTS, is developed and maintained by contractors. Software management techniques have been tried, see USAF TFWC summary in Volume II. They are now being studied in a controlled environment, see ASTROS summary in Volume II, for use in full-scale production later.

Relation to Phase I

This report includes experiences reported in Phase I [LAPA76] (called Range Support there) as well as two new systems. FORTRAN, although not ANS standard, still stands out as the most desirable language. Use of a FORTRAN preprocessor and structured programming techniques at SAMTEC is ongoing. Although no detail data is currently available, user acceptance is favorable and programmers are applying newly learned skills at a rapid rate.

SIMULATOR AND TRAINER

Data on two Simulator and Trainer (ST) systems is summarized in Volume II. Information on both systems was reported by ESD, while one was reported by MITRE. One of these systems (STEM) is in the RFP preparation stage, and the other (TRACALS-VFR) is in development.

Hardware/Software Environment

Both of the ST systems either use or will use commercially available minicomputers. Commercial compilers, operating systems, and support tools are employed.

Languages Used

The following languages are used by the ST systems:

- FORTRAN - 1 (TRACALS-VFR)
- some HOL (probably FORTRAN or JOVIAL (J3)) - 1 (STEM)

FORTRAN is suitable for use in exercise preparation, scenario generation, and data reduction routines in ST systems. Exercise conduct programs, which must be executed in real-time, often necessitate assembly language efficiency. The languages used in ST

systems must support heavy display generation and handling requirements.

Language Selection

The specific HOL used in each of the ST systems was selected by the contractor. STEM requires the use of some HOL, while TRACALS-VFR does not.

The factors which had the greatest impact on the language selection process were:

- off-the-shelf approach (STEM)
- availability of compiler (TRACALS-VFR)
- programmer training (STEM)
- maintainability/reliability (STEM)

Language Standards

In the case of both systems, the contractor was free to choose the language used (as long as some HOL is used for STEM) and no specific language standards were applied. The FORTRAN IV compiler used in TRACALS-VFR adheres to the ANSI standard.

Software Development and Maintenance

The software in both ST systems was or will be contractor developed. Both systems have a central support facility. Air Force personnel maintain the entire TRACALS-VFR system, while STEM will have commercial maintenance for support software. In both systems, Air Force personnel are responsible for making changes to the training exercise programs and for performing other in-house programming tasks.

Relationship to Phase I

In Phase I of the AFSC HOL Standardization Program [LAPA76], data was reported on 21 avionics flight crew simulator and trainer systems. These avionics ST systems differ in experience from the two systems reported here. Nineteen of the 21 avionics systems used assembly language, while two used FORTRAN experimentally. Language selection was influenced in large measure by the fact that Datacraft 6024 series computers have become de facto standard because the system descriptions prepared for vendor bidding specify equipment

configuration in addition to system requirements. Since Phase I, non-standard FORTRAN has proven to be adequate for avionics ST systems.

SUPPORT SYSTEMS

Information pertaining to five Support (SUP) systems, all operational, is presented in Volume II of this document. Four of the systems were reported by SAC/ADXRM, while one was reported by SAMSO.

Hardware/Software Environment

Support systems are characterized by having a single data processing center. Four of the five SUP systems employ large mainframe computers, e.g., three systems include WWMCCS H6000 series computers, two use IBM 360/370 series machines, and one a CDC 6600. All five systems use commercially provided operating systems, compilers, and support software.

Languages Used

The following languages are employed in the five SUP systems:

- FORTRAN - all 5 systems
- assembly language - 3 systems including 1 with incidental use
- COBOL - 2 WWMCCS systems
- JOVIAL (J3) - 1 WWMCCS system

The bulk of the application software in SUP systems consists of programs performing scientific computations. Thus, FORTRAN is well-suited to this environment. FORTRAN is used for such scientific applications as:

- spacecraft event sequences
- spacecraft ephemerides generation
- ballistic maneuvering, reentry vehicle trajectory reconstruction, and display
- data reduction

- telemetry, radar, and optical data analysis
- flight simulation
- intelligence evaluation and reporting

In Support systems, assembly language is used primarily to improve either memory utilization or the performance of critical functions.

Language Selection

In the case of three of the five SUP systems, FORTRAN was employed because of a requirement levied by either the user or SPO. The reasons cited for desiring FORTRAN were:

- suitability for application - 3 systems
- software transportability - 2 systems using ANSI standard FORTRAN
- availability of compiler - 1 system
- programmer training - 1 system

Assembly language is used in Support systems principally to meet processing requirements. One system uses assembly language for maintenance patching.

Language Standards

In the case of the two WWMCCS Support systems, WWMCCS standard COBOL [ANSI68], FORTRAN [ANSI66A], and JOVIAL (J3 as defined by [AIRF67] are used. No data on language standards applied is available for the other three systems.

Software Development and Maintenance

Commercially available system and support software is employed in all five cases. The application programs for the four SUP systems reported by SAC/ADXRM were written by the user, while the contractor for DS&A developed its programs.

Relation to Phase I

This is a newly identified application area. All of these systems are reported here for the first time.

SURVEILLANCE AND WARNING

Five Surveillance and Warning (SW) systems are summarized in Volume II. Four were reported on by both ESD and MITRE; one was reported on by ESD alone. Three of these systems are in development, one is a preliminary study prior to RFP (Request for Proposal) for full-scale development (GEODSS) and one is about to issue its RFP (JSS).

Hardware and Software Environment

Both systems in development use large-scale, commercially available computers; minicomputers are used for radar control. All systems use or plan to use commercially available operating systems, support software, compilers, and assemblers.

Languages Used

The following languages are used by SW systems in development or for the preliminary study:

- assembly - 5 including 3 incidental use
- JOVIAL (J3) - 2
- FORTRAN - 3 including 1 incidental use
- HOL (to be determined) is required for JSS.

All systems except GEODSS have or plan to have major HOL use. JOVIAL (J3) is most frequently used on the primary processor for command and control information development functions.

FORTRAN is used to perform tracking and radar processing/controlling algorithms which require heavy scientific computations. Two FORTRAN compilers have many added features, see COBRA DANE; FORTRAN is used on minicomputers as well as the primary processor, see PAVE PAWS.

Assembly language is used to some extent on all systems especially for time-critical portions of code. It is also used for minicomputers, e.g., COBRA DANE. GEODSS required assembly language to support the high volume of bits manipulated in real time.

Language Selection

Three SW systems were designated Command and Control Systems and so were required by AFR 300-10 to use JOVIAL (J3). A waiver has been requested for one of these systems, i.e., JSS; an HOL (to be determined by the contractor) will be required, as was done for COBRA DANE.

Factors which most heavily influenced the language selection decision are:

- availability of compiler - in cases where an HOL was required (3 systems);
- hardware selection (2 systems);
- overall system design, suitability for application, programmer training, off-the-shelf approach, maintainability, reliability, and user experience - influenced systems using HOLs.
- processing requirements and programmer training (past experience) - influenced the system using assembly language.

Language Standards

The AFR 300-10 [AIRF71] requirement to use JOVIAL (J3) has been invoked for three systems designated as Command and Control; one has requested a waiver. AFM 100-24 [AIRF67] defines the J3 language; the JOVIAL Compiler Validation System (JCVS) was used to validate the UNIVAC (J3) compiler for CONUS OTH. The CDC compiler used in PAVE PAWS will also be validated.

FORTRAN IV is defined by ANS X3.9-1966 [ANSI66A]. The CDC FORTRAN compiler exceeds the standard by the features listed in Volume II (see COBRA DANE); several of these extensions made the language suitable for this application. UNIVAC's FORTRAN compiler is called FORTRAN V to indicate extensions to the standard.

No other language standards were required or used.

Software Development and Maintenance

All systems except the GEODSS feasibility study depend on contractors for software development. Maintenance is performed by Air Force personnel in one case and a contractor in a second; others are unknown.

These systems have long lives, e.g., 10 - 20 years, so cost of ownership is important to life-cycle cost.

<u>Relation to Phase I</u>

This application area is newly identified in Phase II and covers systems not reported on before.

SECTION IV

SUMMARY OF AIR FORCE EXPERIENCE

Up to this point the analysis has focused on each of the individual application areas. In this section patterns across all application areas and specific issues affecting all areas are discussed.

As can be seen in Table I, two systems were reported as Automatic Test Equipment (ATE), seventeen as Command and Control (CC), ten as Communications (COMM), four as Information Management (IM), three as Navigation (NAV), eleven as Operational Flight Programs (OFP), five as Range Operations (RO), two as Simulator and Trainer (ST), five as Support (SUP), and two as Surveillance and Warning (SW).

As Table II illustrates, many organizations contributed to this effort, but MITRE and ESD together provided data on 30 of the 64 systems surveyed. Of the 64 systems, five are in planning, four are in source selection, twenty-two are in development or testing, twenty-eight are operational, and five have segments in various stages of the acquisition process.

Table III itemizes the principal computers used in the Air Force. As can be seen, CDC 6000/7000 and Cyber 70 Series, Honeywell 6000, IBM 360/370, and UNIVAC 1100 Series (or AN/UYK-7) are the large-scale computers most often used. UNIVAC 1600 (or AN/UYK-20), CDC 3000 (Model 3800), and Data General Nova (or Rolm 1601 or AN/UYK-12) are the most common medium-scale machines.

Since each computer comes with at least one unique assembler, over 34 assembly languages are represented in this data (see Appendix I). In addition, several cross-assemblers which run on a large-scale computer and generate code for a smaller computer are used but not tabulated here. This practice is especially common in avionics, missile, and space applications (for operational flight programs).

Language Use and Standardization

From Table IV it can be seen that about half the systems surveyed (30 systems) make or plan to make extensive use of assembly language. Another fourth (15 systems) use it incidentally. A total of 51 out of 64 systems reportedly do or may use assembly language to some extent for application programming; 31 of these 51 systems use assembly language in addition to or in conjunction with an HOL. Five systems have not selected the language to be used.

The high order languages reported and their use are:

ALGOL - used incidentally in one system.

APL - used for numerical applications in two systems.

ATLAS - ATE language used with higher incidence among systems reported in Phase I [LAPA76].

BASIC - interactive language used to some extent in four systems.

CMS-2 - Navy standard language used in inter-service systems.

COBOL - used to some extent by ten systems concentrated primarily in CC, IM, RO, and SUP areas. It is used for batch data management applications and is the primary language for one reported system, MACIMS. Most systems, especially those in the WWMCCS family, required adherence to the COBOL standard, ANSI X3.23-1968 [ANSI68], but extensions are common. The current federal standard X3.23-1974 [ANSI74] differs from the 1968 standard in some areas; AFR 300-10 currently requires the 1968 standard.

DSPL - a special-purpose display language developed for and used in one system, TRI-TAC TCCF.

FORTRAN - most widely used HOL (see discussion below).

HPBASIC - a widely used ATE programming language (reported as most widely used ATE HOL in Phase I [LAPA76]).

JOVIAL - second most widely used HOL (see discussion below).

MUMPS II - a special-purpose interactive language used for one system, AFEES.

SIMSCRIPT - a simulation language used occasionally by two systems.

SPL - a space-oriented language used in one system, DMSP Space Segment.

FORTRAN

Twenty-one systems (one-third of those surveyed) use some version of FORTRAN while four more make incidental use of FORTRAN and two others may use FORTRAN. It is used in all application areas to some

extent, but to a lesser extent in the ATE, OFP, and NAV areas. Much of this use, especially in ST and occasionally in ATE, COMM, OFP and SW, is for programming minicomputers. FORTRAN is used as the primary language for ST, RO, and Support systems. It is used for scientific computations in a batch environment in the SUP, CC, and COMM areas. Minuteman III is a notable exception in which FORTRAN is used for mission-oriented OFP software.

AFR 300-10 [AIRF71] requires use of FORTRAN IV as defined by ANS X3.9-1966 [ANSI66A] or X3.10-1966 [ANSI66B] for advanced numerical applications. (A draft proposed revised FORTRAN Standard is in preparation [ANSC76].) FORTRAN V is the UNIVAC implementation which includes extensions to the standard.

Although many FORTRAN compilers are commercially available (see Table V) and many are used in the reported systems (see Table IV), adherence to ANS FORTRAN X3.9-1966 is largely ignored by Air Force users. Most compilers comply with the standard as a base and implement many extensions, e.g., CDC FORTRAN on COBRA DANE; these extensions are frequently the reason FORTRAN is able to support the application. In a few cases in which application software portability was desired, e.g., Minuteman, restriction to a language subset was required.

JOVIAL

Twenty-two systems (about one-third of those surveyed) use or plan to use at least one version of JOVIAL while three others may use a version of JOVIAL. This use is concentrated primarily in the Command and Control and Surveillance and Warning application areas with a growing number in the OFP area.

Four main versions of JOVIAL are represented:

- JOVIAL (J3), which is defined by AFM 100-24 [AIRF67], is used by ten CC and two SW applications on large-scale computers requiring handling of large data bases. Use in CC systems is required by AFR 300-10 [AIRF71]. Some of the compilers in use have been verified by the JOVIAL Compiler Validation System (JCVS); extensions to or minor deviations from the standard language are common. In two cases (WWMCCS and PAVE PAWS) adherence to the standard was a requirement for acceptance of the compiler.

- JOVIAL (J3B), which includes three dialects J3B, J3B-1, and J3B-2, was designed by SOFTECH initially for the B-1 program [SOFT76B]. It is currently in use in one other program and

has been used by AFAL on the OSC (Operational Software Concept) Program; these are all OFP applications. J3B and J3B-1 programs can be compiled by the J3B-2 compiler.

- JOVIAL J73/I, which was developed by RADC in conjunction with representatives of the user community, is used on one OFP program, DAIS (Digital Avionics Information System), at AFAL and by leased line to AFAL at ADTC. This implements only Level 1 of three defined subsets. Several OFP systems (only one is reported here) are considering use of J73/I. The latest version of the language standard [RADC76] reflects criticisms made since the DAIS version was developed. It is not identical to the language implemented by DAIS compilers, but these differences are being resolved.

- JOVIAL (J4), which is defined by a SAMSO Technical Report, is used by four programs (three of which are CC and one of which is NAV) all of which use the AFSCF (Air Force Satellite Control Facility). Only one J4 compiler on the CDC 3800 at the AFSCF is used; it is similar to J3 but has unique input-output features.

Compiler Availability

Table V itemizes the compilers available on twenty-two of the large- and medium-scale computers used on the systems surveyed. Many of the commercially available compilers, e.g., for ALGOL, BASIC, and COBOL, are not used in the systems reported. PL/I, with three compilers available, is noticeably unused by Air Force weapon and defense systems.

The FORTRAN compilers tabulated frequently reflect more than one version by a specific vendor; this does not include compilers sold by independent software developers. The JOVIAL (J3) compilers tabulated include three that are commercially available, Honeywell 6000, UNIVAC 1108, and CDC Cyber implementations. In addition the Air Force owns J3 compilers for Honeywell 6000 (the JOCIT compiler), Hughes 118, IBM 360/370 (developed by System Development Corporaton), and AN/UYK-7 (2 implementations). Two J3 cross-compilers, one from 360 to 4 Pi CC-1 and one from AN/UYK-7 to AN/UYK-25, are also AF owned. All J3B and J73 compilers are AF owned.

Compiler building tools have been developed, including one for JOVIAL (J3) called JOCIT (JOVIAL Compiler Implementation Tool) which has been used to develop one compiler (for WWMCCS systems) to date. The other tool is CWS (Compiler Writing System) derived from SPLIT (Space Programming Language Implementation Tool), which was

originally intended for implementing versions of SPL at SAMSO. To date it has been used to write one SPL compiler (for DMSP) and to implement HAL/S and OPAL [FELT76]. Claims about the ability of Computer Writing System to implement compilers for a wider range of languages, e.g., FORTRAN and J73/I, have not been verified.

Factors Affecting Language Selection

The factors affecting the language-selection decision of each of the sixty-four systems reported are tabulated in Table VI. See DATA SUMMARY for a description of the categories. The relative importance of each decision agent is discussed below:

1. Requirement: Air Force directive - nine systems, concentrated in CC, IM, and SW application areas. In eight cases, AFR 300-10 [AIRF71] was seen as the principal reason for the selection of JOVIAL (J3), i.e., all systems were considered to be Command and Control. One system, i.e., MACIMS, was governed by AFM 171-100 [AIRF74], Air Force Automated Data Systems (ADS) standards Volume I of which references AFR 300-10 and itemizes specific languages feature requirements for COBOL.

   Other systems may have been required by AFR 300-10 either to use JOVIAL (J3) for command and control applictions or to use FORTRAN for numerical applications, but respondents did not see this regulation as the primary reason that the language was selected.

2. Requirement: User/SPO - eleven systems in all application areas except SW and including primarily CC, RO, and SUP. All four systems using the AFSCF (Air Force Satellite Control Facility) and many TAC (Tactical Air Command) and SAC (Strategic Air Command) systems which use HOLs required the selected language, primarily JOVIAL. Most RO systems especially at SAMTEC, are required to use FORTRAN.

3. Requirement: User/SPO (class of language) - nine systems required use of any HOL or choice of a limited number of HOLs. These are primarily recent acquisition programs, e.g., JTIDS/ASIT and the B-1 Bomber. In all cases, the contractor will choose (or has chosen) the specific language. These systems are in all application areas except IM, NAV, and SUP, with the majority in the OFP area.

4. Developer discretion: Contractor - thirty systems including nine from the previous category. These systems are in all

application areas except SUP. The areas of CC, OFP, ATE, and NAV include the majority of these systems.

5. Developer discretion: User - one system developed and operated by the user.

6. Developer discretion: Air Force/other agency - two systems which are planning or feasibility studies and one agency, AFAL, which has several programs ongoing.

The relative importance of each factor underlying the language decision follows:

1. Overall system design - four systems primarily in recent acquisition programs for which source selection has not been made.

2. Suitability for application - thirty-three systems in all application areas except ATE (where ATLAS use was discussed in Phase I [LAPA76] as being most suitable to the ATE application). For application areas where suitability was deemed important, it is the principal factor. Notable exceptions to this trend are the OFP and COMM areas; here hardware selection is the principal criterion.

3. Procesing requirements - ten systems primarily in the COMM and OFP areas where critical time and memory constraints are prevalent. Some of these systems also indicated that the language chosen was well-suited to the application.

4. Hardware selection - twenty-two systems in the ATE, CC, COMM, OFP, and SW application areas. In these systems language considerations depended on the hardware chosen, e.g., what translators came with them.

5. Off-the-shelf approach - eight systems scattered among the application areas have required (or are requiring) support software, especially compilers, to be available off-the-shelf. Most of these are recent acquisition programs which are trying to reduce cost and risk by procuring software which has previously been used.

6. Availability of compilers - nine systems scattered among the application areas, but predominently in SW. Together with the previous category, 18 systems have been influenced by the availability of existing (and in most cases tested) support software.

7. Programmer training - seventeen systems scattered among all application areas except Communications with the heaviest emphasis in RO. In some cases, e.g., OFP, ease of training is seen as a benefit of use of any HOL, as opposed to assembly language, not just a particular language.

8. Maintainability/reliability - eight systems primarily in NAV, OFP, ST, and SW areas. Again improved reliability is seen as a benefit of using HOLs in general, e.g., for OFP. In one case, ASTROS, it is hoped that use of a FORTRAN preprocessor will result in more easily maintainable code.

9. Software transportability - eleven systems to some degree in all areas except IM, OFP, ST, and SW. Many of these systems are part of the WWMCCS family, e.g., NORAD and PACOM C4, or are based on a prior system, e.g., TIPI-TERPE. Application software for these systems is being converted and users are discovering differences in language versions by experience. The requirement to transfer software shows up more strongly in this report than it did previously in [LAPA76].

10. User experience - ten systems primarily for TAC, SAC, and MAC, in the CC, IM, RO, and SW areas.

11. Standard language - three systems which planned to transport software before any development began.

12. Reuse of compilers - six systems in the CC and NAV application areas. WWMCCS, AFSCF, and TIPI systems planned to use existing compilers and support software.

13. Software engineering support - two systems using FORTRAN preprocessors.

Although no formal language selection process is evident from the data collected, certain factors affect the selection of each major programming language more than others (See Table VI).

Assembly

Assembly language is most often selected by contractors in order to meet processing requirements and because it is frequently the only language supported on the hardware selected. Assembly language has been considered suitable to the applications for which it is used, e.g., Communications and OFP, and assemblers are available off-the-shelf.

COBOL

COBOL is usually either required by the user or SPO for use by a contractor or is selected by the user for internal software development support. User experience and COBOL's suitability for data processing applications are cited most often as reasons for its selection. Programmer knowledge of the language, transportability of existing software, and the fact that it is a standard language are other reasons for using COBOL.

FORTRAN

FORTRAN is usually required by the user or SPO. It is sometimes selected by the contractor, frequently in response to a user or SPO requirement for a class of language. FORTRAN is considered suitable to the applications for which it is used (primarily minicomputer or batch scientific ones), is known by a large base of programmers, and can be used to write substantially transportable software. Availability of compilers, hardware selection, and user experience support FORTRAN's role as a widely used language. In addition, software engineering support tools, e.g., FORTRAN preprocessors, are currently available, but have a small influence (when viewed against the total base of Air Force experience).

JOVIAL (J3)

JOVIAL (J3) is usually required because of AFR 300-10 [AIRF71]. Otherwise it is required by the user or SPO primarily because of user experience (e.g., at SAC and TAC) with the language. Suitability to the command and control application and the ability or desire to transport software written in J3 are other less significant reasons for using J3.

JOVIAL (J3B)

JOVIAL (J3B) has been selected by contractors in response to a user or SPO requirement for an HOL. It is considered suitable for avionics OFP applications. Program maintainability/reliability and ease of programmer training have improved or are expected to improve by using an HOL instead of the traditional approach which is to use assembly language.

J3B was developed initially because complete specification for J73/I, which was desired, was not available. Subsequent SPO requirements for a JOVIAL-based language resulted in enhancements to J3B, leading to J3B-1 and J3B-2.

JOVIAL (J4)

One JOVIAL (J4) compiler is in use; it is required by the user for all development programs using the Air Force Satellite Control Facility (AFSCF). J4 is considered to be suitable for the application.

JOVIAL 73/I

JOVIAL 73/I has not been used by any production-oriented system. It is being used at AFAL on a variety of programs [TRAI76]. The Level I subset of the full language JOVIAL 73 was tailored specifically to the avionics OFP application.

HOL

Several new acquisition programs are requiring use of a high order language, but not any specific one (to be chosen by the contractor). The decision will be based primarily on the overall system design and off-the-shelf availability of support software; software maintainability/reliability is a factor leading to requirement of an HOL.

Development and Maintenance

Table VII displays the organizations responsible for the development and maintenance of the reported systems. Contractors perform most application and support software development; some of the Major Commands, i.e., users, develop their own software.

Much less was reported about software maintenance responsibilities, partly because this decision has not been made for seven systems surveyed. There is a movement away from contracted maintenance. Users and other Air Force agencies, such as CCPC and CCTC, are taking over maintenance duties. Nine systems reported are either part of the WWMCCS family or are using WWMCCS-compatible systems. A mechanism for developing and maintaining shared WWMCCS software is in use.

AFLC provides maintenance support to ASD avionics programs, especially for electronic warfare subsystems. How much programming language responsibility this involves is not clear from the data reported.

SECTION V

CONCLUSIONS

The conclusions presented in this section are based on the data which has been collected over the past two years by the AFSC HOL Standardization Program team. The process by which we arrive at the Approach to Standardization later in this section is shown pictorially in Figure 2.

Review of the data clearly indicates that many factors affect language selection for an acquisition, one or more of these factors sometimes yielding very powerful influence in particular cases. Review of the situation today, however, reveals that some form of standardization is desirable and can be achieved within the Air Force at a cost far less than the savings to be gained by standardizing. The critical choice will be the form of the standardization -- that is, the specific policies and implementation plans to be adopted by the Air Force. A successful standardization activity must be characterized by implementation guided by existing practices and capabilities and by risk reduction for acquisition programs. This means that certain conditions must exist or be created at the user's level in order for standardization to work. These conditions are the same as the answers to the question "Why have organizations used particular languages?"; they are:

- availability of compilers
- suitability to application
- useable result (object code) on available/selected computer
- experience supports/favors use of the language
- industry supports the language
- support tools for the language and/or its compiler/assembler are available
- in the case of a new language being adopted by an organization (e.g., transitioning from assembly language to FORTRAN) there has been adequate time for motivation to transition to develop from within rather than as a response to a formal requirement imposed from without.

INPUTS (PHASES I & II)		ANALYSIS (PHASE II)	OUTPUTS
high order languages	→	identification of use of HOLs and reasons for success or failure	
acquisition practices	→	identification of factors affecting language selection	
application areas and functional requirements	→	identification of commonalities and unique requirements	
assembly language	→	determination of usage and cases where considered essential	Approach to Standardization ↓ Policy ↓ Implementation Plan
standards - language - management - formal/informal	→	identification of attempts - standardizations and reasons for success for failure	
policies and intiatives (AF and DoD)	→	determination of existence, usefulness, impact	
technology trends	→	analysis of impact on language use and selection	
software development engineering practices	→	use and relationship to languages	

Figure 2. Evolution of Standardization Approach

These conditions can be re-stated in the form of near-term issues which raise obstacles to standardization. For example, a user (SPO or Major Command) may desire to use a particular, non-standard language because of familiarity with it or an existing investment in hardware, software, programmers, or interfaces. A user will normally shy away from using a new compiler and its associated support software because of a perceived additional risk, even if the language being supported by the compiler is called a standard language. Contractors may pressure the user toward use of the contractor's tools, since they are known quantities from the contractor's point of view; the user may acquiesce because of an inability to counter the contractor's arguments and because of a habit of reliance on the prime contractor, a habit which is sensible in many cases, since the prime knows and can apply technology in his area of expertise, and, therefore, a habit which can be replaced only if sound, low-risk alternatives are available.

Although arguments for standardization have been presented many times in many places before, a summary of arguments for and cumulative benefits expected from standardization follow <u>and are based on the information gathered over the past two years</u>: that is, they are inductive statements, not claims deduced from an untested premise. Next in this section appear summaries of the conditions required for standardization, progress to date, and potential obstacles. Finally, the recommended approach to standardization is presented, motivated by the material of this section which precedes it.

<u>Why HOL Standardization?</u>

As mentioned in the INTRODUCTION, the Department of Defense is seeking to "reduce the proliferation of HOLs in Defense Systems and to insure control of those HOLs which are approved" [DoDI76]. The list of DoD approved high order programming languages is subject to change, even after a "minimal set" of HOLs has been established. The Air Force must not only meet the immediate requirements and responsibilities imposed by this DoD policy, but must prepare itself to respond to future changes in allowed languages. At the same time the Air Force must continue to conduct the business of system acquisitions, upgrades, and maintenance.

The immediate benefits sought from a standardization activity are the reduction of development and maintenance (i.e., life-cycle) cost, time, and risk while satisfying the technical requirements of each system. Program Offices (POs) and other users should be assisted in making cost-effective decisions on language choice and, by extension, in choosing support software. Standardizing the programming language

used can stabilize the programming environment, thereby providing a basis for application of additional technologies and leading to other indirect benefits.

Cumulative Benefits

Although cost information in the form of dollar figures was and is not available, the Air Force and other government programs studied provide strong evidence that long-term benefits can accrue from standardization of high order languages, in much the same way as they have from the CORAL 66 standardization in the United Kingdom. Specific cost-related factors can be identified in Air Force programs; these factors can be dealt with through an effective standardization activity to reduce cost and risk. Reduced compiler and support software costs, better compilers with more reliable attendant software, reduced programmer training costs, encouragement of a standard programming environment which can lead to better documentation and improved management practices, reuse of software support tools -- all of these can be achieved with resulting, measurable cost reduction. Other factors are also significant even though the result of manipulating them cannot be measured directly in dollar savings. Some of these are improved software reliability through reduction in the use of assembly language, extended programmer experience, and industry support.

A standard language provides a medium for communication and documentation. The self-documenting nature of an HOL, compared to assembly-level languages, can lead to improved software maintainability and reliability.

Increased use of a language encourages development of compilers and software tools for that language which, over time, can enrich the programming environment. As this software is tested through use, bugs can be identified and eliminated, leading to improved support software quality. Improved compile-time and execution-time performance is another potential benefit of greater use of specific compilers. Reuse or transportability of support software, especially a total programming environment, results in reductions in life cycle cost and time.

The existence of compilers adhering to a language standard can lead to greater success in developing application programs which are readily transported between computer systems. Because reuse of application software is dependent on factors in addition to language, such as operating system interface, it is not widely attempted now except for those situations where investment in existing programs or

development schedule dictate program transfer. At a minimum, decreased conversion costs can result.

Language standardization also leads to increased user, e.g., Major Command, experience with a language. Increased use of a language leads to accumulation of user software in that language and of in-house programmers knowledgeable with the language. The user develops an appreciation for the software development and maintenance processes as they are visible through the language. For example, SAC has been insistent that, in systems which they will maintain, JOVIAL be used because of SAC's extensive experience with it, existence of a significant programming pool for it, and suitability of JOVIAL for their applications. Although not stated by SAC, one reasonably concludes that these are arguments based on SAC's desire to reduce or minimize cost and risk. This kind of experience-based argument encourages one to consider extension of SAC's situation to a broader base by adopting a standard language, in particular JOVIAL, for a class of suitable applications.

Continued use of a standard language creates a pool of programmers with expertise in the standard language both within and outside DoD. Costs to educate personnel will decrease as skills are reused instead of relearned.

A standard language that is widely known, accepted by users, and supported by government and industry enables the development of software, especially application software, to proceed somewhat independently of the rest of the system, yielding reductions in cost, time, and risk of software acquisition. The government, in particular the Air Force, has more than sufficient motivation to encourage adoption of standard languages. Industry will need to perceive self-serving benefits in order to support standard languages; this must be, and can be, accomplished by providing appropriate incentives, to insure that government and industry are working toward the same objectives, albeit for different reasons.

<u>Conditions for Standardization</u>

The study of Air Force and other government programs and experiences presented in the literature argue that the following conditions must be in order to achieve HOL standardization:

- The language must be suitable to the application, i.e., it must satisfy the system's technical requirements. Frequently, any one language of a class of languages with similar capabilities is sufficient. Applications such as

Communications or Operational Flight Programs have more critical processing requirements than other Air Force application areas and, therefore, may require an HOL with more technologically advanced features.

- Compilers for the language must be available for or with proposed hardware. They can be provided off-the-shelf (existing and at low cost) by industry or government or they can be readily developed with minimal risk in terms of time and money. Efficiency of compiler-produced object programs is most important for applications, such as COMM and OFP, which have critical processing requirements.

- Software tools needed to support programming in the language must be available for or with proposed hardware. These tools range from editors to software engineering-related tools. They can be provided off-the-shelf by industry or government or they can be developed at minimal risk.

- Programmers skilled in the use of the language must be available. Such skills can be acquired from experience within or outside DoD or the Air Force or from programmer training programs.

- Language must be mature enough so that problems have been identified and eliminated. At least four years of use is generally required before a language standard can be frozen.

- Commitment to language standardization in terms of time and funding by the Air Force, including POs and users, and support from Air Force contractors is required.

Progress to Date

A wide variety of computer programming languages is used on systems acquired by Air Force and other government agencies. High order languages, rather than or in addition to assembly level languages, are often used, especially on recent acquisition programs. Progress toward HOL standardization has been made; in many organizations use of a particular language is preferred or required. For example, TACPOL is preferred for Army systems, CMS-2 is required for Navy shipboard systems, JOVIAL (J3) is required for Air Force command and control systems, FORTRAN is preferred for SAMTEC range support systems, and HAL/S is preferred for NASA space shuttle applications.

Some of these preferences have evolved into _de facto_ standards, e.g., FORTRAN at SAMTEC. In other cases, the language has been standardized by requiring its use, controlling its compilers, and monitoring its extensions, e.g., CORAL 66 in the United Kingdom. In particular, the Air Force generally has been moving toward use of standardized high order languages.

Use of COBOL and FORTRAN is widespread in the Air Force primarily for data processing and scientific applications, respectively. (They are used by over one-third of the systems surveyed.) Compilers based on past COBOL [ANSI68] and current FORTRAN [ANSI66] language standards are used. They are provided free or at minimal cost by vendors who market them commercially. They contain vendor-specific extensions and implemention dependencies which complicate transfer of programs between systems. For some applications, e.g., COBRA DANE, it is believed that these non-standard features are required to do the job; for others the standard language is adequate, e.g., Range Operations. For some applications, e.g., USAF Tactical Fighter Weapon Center, attempts to limit the use of extensions failed under normal development pressures, especially time; in a few cases, e.g., Minuteman III, a language subset was required and enforced by the compiler.

Use of JOVIAL-based languages is widespread in the Air Force primarily for avionics operational flight programs, surveillance and warning, and command and control applications. (They are used by one-third of the systems surveyed.) AF-owned compilers implement four major versions of JOVIAL (J3, J3B, J73/I, and J4); in addition, three versions of J3B and several extensions to J3 are represented. Three JOVIAL (J3) compilers are available commercially for the Honeywell 6000, UNIVAC 1108, and CDC Cyber systems. In addition, five J3 compilers for four systems, including JOCIT JOVIAL for the Honeywell 6000 and two J3 cross-compilers, were developed or modified at Air Force expense and are Air Force-owned. There are five different JOVIAL (J3B) compilers all hosted on the IBM 360/370 and all supplied by one vendor [TRAI76]. There are five JOVIAL J73 compilers (for three hosts and five target machines) implementing various subsets of the full J73 language, primarily J73/I, with other cross-compilers in development [TRAI76]. These compilers are supplied by three different vendors. These are all developed at Air Force expense.

In the United Kingdom CORAL 66 has become the standard language for weapon and defense systems in the Ministry of Defense (MoD). Compilers for at least 45 machines are used. These have been developed by potential vendors, not funded by MoD, and have been accepted for MoD use after passing an assessment test. Passing this

test assures that the compiler implements the language as specified; a review of the documentation checks for extensions to the standard. The availability, small size, and relatively low cost of these compilers has led to the use of CORAL 66 in industrial applications both in the UK and the US.

As reported in [LAPA76] ATLAS is a commercial standard for Automatic Test Equipment applications and is proposed to be used extensively for Air Force automatic test equipment applications, especially avionics-related systems.

Investment in programs written in, support tools for, and personnel familiar with COBOL-68 (1), FORTRAN-66 (2), and JOVIAL (J3)(3) exists within the Air Force. In addition, industry-wide and broad government support exists for FORTRAN and COBOL. Progress is being made in determining compliance of compilers with standard specifications, e.g., COBOL Compiler Validation System (CCVS for COBOL-68 and COBOL-74(4)), JOVIAL Compiler Validation Systems (JCVS for J3 and J73/I), and the nucleus of a FORTRAN Compiler Validation System (FCVS for FORTRAN-66 and proposed FORTRAN-76(5)) [LAPA76]. Extensions to language standards are still permitted in the Air Force. Versions of JOVIAL and subsets of versions are still being developed [TRAI76]. Introducing COBOL-76, FORTRAN-76, and an upgrade for J3 into Air Force acquisitions poses transition, timing, funding, and training problems which must be addressed by policy and plans for implementation of Air Force-wide HOL standardization.

Potential Obstacles

If a radical change were to occur in the near future in either the DoD or industry positions on high order languages, it might necessitate almost total reevaluation of the Air Force position. For example, DoD might impose a requirement to use a new, common HOL prematurely from the point of view of the Air Force; that is, DoD might require Air Force systems to use a new, common HOL before the conditions conducive to standardization, such as those identified earlier in this section, had developed. Also, industry might adopt a new position on PL/I and begin to make compilers generally available. Neither of these occurrences seems likely; on the other hand, there seems no way to plan ahead for such occurrences.

---

(1) ANSI standard of 1968 [ANSI68]
(2) ANSI standard of 1966 [ANSI66A]
(3) Air Force standard, AFM 100-24, of 1967 [AIRF67]
(4) ANSI standard of 1974 [ANSI74]
(5) draft ANSI standard of 1976 [ANSC76]

More likely are changes in processor architectures as microprocessor-based computer systems become more prevalent. Several scenarios can be imagined based on current practice. For example, a re-emergence of assembly-level language as the basic computer programming language might occur if industry should decide that it is not profitable to invest in compilers or compiler-builders for the new architectures. Or users might find that the standard language is no longer able to take adequate advantage of architecture features. The technical obsolescence of a standard language is not of concern to the user until it impedes his ability to get the job done. These obstacles can be overcome by appropriate planning and investment in advanced development activities.

Another obstacle could be the cost to the government of controlling standard languages. This obstacle will prevent standardization in two cases:

- if the cost of control is estimated or perceived to exceed or match the total life cycle cost savings achieved by standardization;

- if the estimated or perceived front-end cost (capital investment to start) exceeds the amount of available monies.

Again these cases do not appear to pertain.

Poor user acceptance of a standard language could also undermine a standardization program. This would certainly occur if the choice of language presented a radically new situation to the user; exceptions to the use of the standard would abound and would reduce total life cycle cost savings achieved through standardization to next to nothing. This should not happen, however; the approach to be proposed specifically takes this into account.

Finally, there are technical problems which are less significant, such as the adequacy of semantic specification of a language. This is minor because, although no formal descriptive language is acceptable today, a number of semantics-validation tools exist and can be used.

Approach to Standardization

Deduced from the premises which have been inductively generated over the past two years of the program are the following guidelines for an approach to standardization:

(1) Any standardization plan must be reviewed and revised as appropriate at least yearly in order to accommodate changes in the environment caused by DoD policy, industry initiatives, or technology trends.

(2) Users must in essence be able to respond to the plan from the position in which they are at the time the plan is put into effect; this means that either:

(a) the user need only incorporate new tools or technologies into his existing procedures in order to comply, or

(b) if the user needs to move to a new language, he must have adequate guidance and incentive -- i.e., he must see a success path to compliance.

(3) Existing languages must be adopted as standards and there must be compilers and support tools available or they must be obtainable at acceptable cost; new languages must be programmed for future adoption, at a time when they can both do the job required and be adequately supported.

(4) Current and anticipated industry and DoD initiatives should be taken into account as far as possible in the initial formulation of a plan.

(5) Technology trends should be taken into account as far as possible in the initial formulation of a plan.

<u>Elements of an Air Force HOL Standardization Policy</u>

The following elements of a policy satisfy the approach, address the potential obstacles, foster the conditions, build upon the progress to date, and derive from the information presented in this report.

(1) Languages and language issues:

(a) ATLAS: this language is an industry standard for automatic test equipment which has been recommended both by this program and DoD memorandum as a military standard.

(b) COBOL: this language is used in the Air Force, especially for command and control systems, and is well suited to the applications for which it is used; no immediate replacement is available. COBOL should be supported and controlled, phasing from

COBOL-68 to COBOL-76 as required by the National Bureau of Standards in December 1975.

(c) FORTRAN: this language is widely used in the Air Force and has no acceptable immediate substitute. Air Force applications use a variety of extensions to FORTRAN-66; the proposed FORTRAN-76 satisfies some, but not all, of these language requirements, e.g., bit manipulation and full structured programming facilities.

Reducing the proliferation of versions of FORTRAN can be accomplished in several ways: by restricting Air Force users to FORTRAN-66, by adopting and controlling a set of Air Force accepted extensions to FORTRAN-66, or by phasing into the next ANSI FORTRAN standard (perhaps a finalized version of FORTRAN-76). These alternatives must be accompanied by support and control of the adopted version of FORTRAN. For some applications, such as simulator and trainer, it may not be cost-effective to restrict contractors to any predetermined version. If such latitude continues, the Air Force must still monitor and approve proposed dialects.

(d) JOVIAL: many versions of JOVIAL are in use for command and control and avionics applications. JOVIAL (J3) is most widespread, having the largest user base, the most compilers, and the most support software. JOVIAL J73/I is a modernized version of (J3) with support developing in the avionics OFP area. One reasonable approach is to support (J3) and J73/I, with eventual (3-7 years) phaseover to J73/I, enabling new systems to take advantage of the technological improvements in J73/I. Another approach is to allow users with programs written in J3, that is, command and control systems, to continue using J3; life cycle cost considerations, especially avoiding the need to maintain programs in both languages during the transition period, make this approach attractive.

(e) Assembly languages: many assembly languages are widely used in the Air Force. The evidence indicates that it is not reasonable to preclude its use in the near future, especially in avionics, communications, and electronic warfare OFP applications; its use may even grow for a time as more microprocessor-based embedded computers are put to use in the Air Force. Standards and guidelines for the use of assembly language are needed and can be provided to insure adequate visibility during development and later maintenance; techniques such as encapsulation and structured design should be employed.

(2) Application areas and issues:

(a) Automatic test equipment: this area now uses a variety of languages; with appropriate support it can move toward a standardized ATLAS, achieving almost exclusive use of ATLAS, [ARIN75] plus Air Force extensions, in the next 3 to 7 years.

(b) Communications: this area now uses assembly language. Evaluations of the effectiveness of existing high-order languages [SOFT76A, DREI76] for communications applications indicate that several are adequate, e.g., J73/I, PASCAL, but none are outstanding. Design of a communications oriented language (COL) [BBN 76] is being sponsored by DCA/DCEC, but results will not be available in the near future. One approach is to adopt an existing, adequate language, e.g., J73/I, as an interim standard (supported by assembly language where necessary) in order to gain experience with HOLs in this area. Another alternative includes allowing exclusive use of assembly or macro-level languages until a superior high order programming language emerges.

(c) Command and control: this area uses JOVIAL, FORTRAN, COBOL, and assembly language. The functional application of these languages is reasonable; they should all be supported in the near term, except that most, if not all, assembly language use could be eliminated. JOVIAL (J3) should be the basic standard, J73/I should be allowed. In 3-7 years J73/I could become the new standard. Alternatively, CC systems could continue using J3 if life cycle cost considerations favor this option (see discussion of JOVIAL under Languages and Language Issues).

(d) Information management: the area is related to command and control. Assembly language is not needed; arguments under (c) apply.

(e) Navigation: this area includes elements of operational flight programs and command and control. Assembly language may still be needed in addition to the standard languages discussed in (c).

(f) Operational flight programs: this area includes avionics, electronic warfare, space, and missile systems and subsystems. Most applications use assembly language, but (J3B) and J73/I are beginning to be used. A viable approach is to standardize on J73/I with assembly language support as required.

(g) Range operations: FORTRAN is the principal HOL used; assembly language is also used. FORTRAN should continue to be supported, assembly language use could be minimized or eliminated.

(h) Simulator and trainer: Assembly language is principally used but FORTRAN has been gaining acceptance. Use of FORTRAN should be encouraged, with eventual elimination of assembly language. The successful use of FORTRAN depends on the availability of suitable language features not in FORTRAN. (See discussion of FORTRAN under Languages and Language Issues.)

(i) Surveillance and warning: this area is related to the command and control area. Most functions can be performed in JOVIAL, but some time-critical sub-functions may require assembly language support.

(j) Issues affecting all areas: assembly language is frequently used for time-critical sub-functions, to minimize storage space required, or to perform executive functions in conjunction with JOVIAL, FORTRAN, or COBOL. Executive functions can be performed in J73/I; as better J73/I compilers become available the need to use assembly language for time or space-critical functions may diminish. Thus, it is viable to move toward (J73/I) as a replacement for assembly language.

(k) Continued use of non-standard languages: enforcement of standard languages is required on future systems. Systems now in development or operation should not change languages. However, if upgrades to existing systems are acquired, life cycle cost justification for continued use of non-standard languages would be required. Similar justification for use of non-standard languages on new systems would be required. Critical life cycle cost factors include existing support software base and maintenance facilities.

(3) Software Engineering Support:

Language extensions, e.g., extensions to FORTRAN, are in use; in the simulator and trainer area, for example, extensions to FORTRAN-66 are required and others are desired. Some extensions would be provided by FORTRAN-76 but not all. In order to effectively control its use of FORTRAN, the Air Force should establish a standardized set of Air Force-approved extensions or monitor and approve specific dialects as required by individual application areas (see discussion of FORTRAN). Language extensions to JOVIAL are frequently used or desired also, but this is a somewhat different problem since the Air Force defines JOVIAL in the first place. In any case, language extensions should be handled by a designated language control agent via a language control facility (see next subsection (4)). Immediate extension of JOVIAL, where required, would be implemented via macros or assembly language; commonly required extensions would feed back into an update of the standard definition of the language.

Preprocessors are being used more and more to support structured programming in HOLs (see Appendix IV for examples); in the Air Force they are available and have been used for COBOL, FORTRAN, and JOVIAL. Over two hundred are available for FORTRAN alone; however, only a few of these have actually been used in the Air Force including S-FORTRAN, one of the best of them, at SAMTEC. STRUCTRAN-1 is available from RADC. When such preprocessors prove themselves in use, the Air Force should select one (per HOL) and standardize it, since each preprocessor presents a new language to the user. Eventually, all preprocessors should be eliminated and the new syntax incorporated into the language standard.

Support tools such as compiler validators (e.g., JCVS), program verifier's (e.g., JAVS), and compiler building tools (e.g., JOCIT) are a formidable and integral part of standardizing HOLs and their use. These support tools should be developed/provided/applied through the Language Control Facilities (see next subsection (4)).

(4) Language Control Facilities:

(a) Requirement: for every language adopted as a standard by the Air Force, i.e., ATLAS, JOVIAL (J3), JOVIAL J73/I, COBOL, and FORTRAN in the short term, a Language Control Facility (LCF) is needed. This goes beyond the DoD requirement of an LCF for J3, J73/I, COBOL, and FORTRAN per DODI 5000.31. The scope of each LCF will depend on the language supported.

(b) Language responsibilites: at a minimum the LCF would be responsible for maintaining the baseline language standard, monitoring and controlling language extensions, testing compilers (or arranging to test) for compliance to current standards prior to acceptance for use on Air Force systems, and maintaining a data base on language use and related issues.

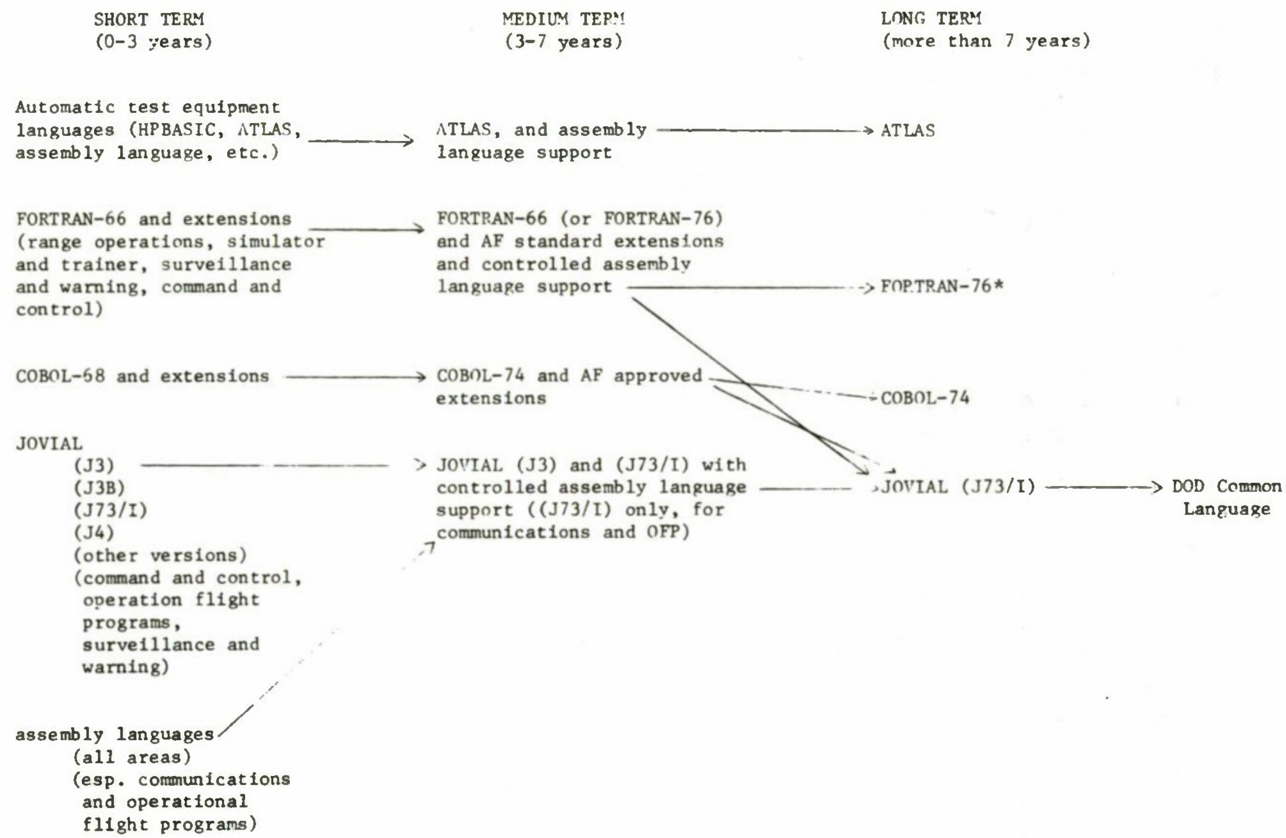
(c) Management responsibilities: guidance to Program Offices (POs) during software planning, acquisition, and maintenance must be provided. Expertise on contractual requirements for contractor compliance with standards and reporting of experience could be concentrated here.

(d) Technical responsibilites: a full-blown LCF could maintain and supply standard compilers and possibly other support software directly to POs, becoming a complete Software Standards and Support Center.

(5) Overview of Standardization Activity:

Figure 3 shows an overview of the transitions to standard languages in the Air Force. Non-standard languages, such as (J3B), would continue to be used so long as total life cycle cost-effectiveness could be demonstrated.

An important part of any standardization activity adopted by the Air Force will be annual policy review. This is necessary in order to account for technology changes and mission reorientations, perform assessment of and incorporate refinements into implementation plans, monitor life cycle cost effectiveness of policy and adjust it to maximize cost benefits, and to maintain responsiveness to DOD Directions and initiatives.



*or new ANSI FORTRAN standard plus AF approved extensions

Figure 3. Transitions to Standard Languages

APPENDIX I

OTHER GOVERNMENT AGENCIES

In pursuing Air Force experience with software acquisition, particularly as it relates to programming languages, it became clear that the other Services and other government agencies faced similar problems. Some data has been collected from non-Air Force sources, although only the NASA inputs are considered substantive enough to accurately represent the contributing agency. The following subsection summarizes what each agency reported or what was gleaned from available material; Volume II contains greater detail on Army, NASA, and FAA experience.

NASA

Introduction

The information summarized here, and given in detail in Volume II of this report, was gathered from the following NASA centers:

John F. Kennedy Space Center (KSC), Florida

G. C. Marshall Space Flight Center (MSFC), Alabama

Wallops Flight Center (WFC), Virginia

Flight Research Center (FRC), California

Goddard Space Flight Center (GSFC), Maryland

Johnson Space Center (JSC), Texas

Lewis Research Center (LRC), Ohio

Jet Propulsion Laboratory (JPL), California

As is evident from the amount of material contained in Volume II, the NASA centers were most cooperative in supplying information. The summary offered here, which includes interpretation and conclusions, represents the understanding of the AFSC HOL Standardization Task team members and has not been reviewed by the various NASA centers.

Summary

NASA is involved in all of the application areas relevant to the Air Force with emphasis on space systems, including missiles, manned spacecraft, deep space unmanned vehicles, space communications networking, orbiting satellites, and the support systems for these.

Software of many types is used, but the focus of programming effort is to install a correct program aboard a space-borne object. Many programming languages are used for a great variety of computers.

FORTRAN is the principal language used for ground-based scientific computation. Space-borne programs are generally written in assembly language (that of the space-borne computer) but a notable exception is the use of HAL/S (a high order programming language) for on-board Shuttle computer programs.

Each center operates somewhat autonomously on a project basis, with cooperation between Centers occurring as required on each program. This mode of operation has led to standardization of computer programming languages by Center when a tangible benefit is perceived. To date, standardization across centers has rarely occurred and has not been imposed by NASA Headquarters. Unless closer ties among Centers develop or become required, it is unlikely that language standardization across Centers would be beneficial.

Languages and Computers Used

Table VIII summarizes the use of languages and computers at the various NASA centers. The language most used at a Center, where known, is underlined.

Conclusions of NASA Experience

Most of the information reported by the NASA Centers is not remarkable, since it largely reflects Air Force experience with respect to computer programming languages. For example, assembly language is used for time critical code and for fully accessing hardware features. FORTRAN is widely used and accepted. Language selection is influenced by factors such as hardware, vendor preference, and suitability to application. Also structured programming technology is increasingly being used, especially by using preprocessors at present.

Table VIII

NASA Computers and Languages

Center	Application Areas	Computers	Languages
GSFC	Test & Evaluation, Scientific & engineering Administrative, OFP, CC, COMM, RO, ST	over 300 from IBM 360/95 to DEC PDP-8 including XDS, UNIVAC, Honeywell, CDC, Amdahl	assembly (1) FORTRAN IV FORTRAN V FORTRAN II CS-1 BASIC COBOL PL/I GPSS APL
JSC	OFP support	IBM 4Pi AP-101 IBM 360	HAL/S
LRC	OFP, ATE, simulation	CDC 6400	assembly FORTRAN
JPL (DSN)	CC, COMM, RO, ATE, ST, Logistics	300 computer-based system planned: 5 med scale 225 minis rest microcomputers	MBASIC
JPL (M J/S)	OFP	special purpose mini UNIVAC 1108	assembly
JPL (MCCC)	CC (space exploration)	UNIVAC 1530 & 1219 IBM 360/75s	assembly FORTRAN SFTRAN (1) PDL (2) MPL (1,2)
KSC	ATE, RO	RCA 110A DCD 160G HIS-635 IBM 360/50	assembly ATOLL FORTRAN COBOL GOAL (3,4)

Table VIII (Concl.)

Center	Application Areas	Computers	Languages
MSFC		UNIVAC 1108 IBM 360 RCA 110A	FORTRAN COBOL assembly GOAL (3) ATOLL I/II (4) MARSYAS/ MARVES BASIC APL, PL/I (5) FORTRAN preprocessor (1,5)
WFC	OFP, CC, COMM, RO, ST	HIS-316/716 HIS-625/635 EMR 6130 NOVA 1220 HP 2108A RCA 4101 UNIVAC 1212/9300	assembly FORTRAN BASIC COBOL IDS
FRC	OFP simulation	CDC CYBER 73-28	FORTRAN COBOL assembly

(1) used for structured programming
(2) software design tool
(3) under development
(4) developed at or responsible for
(5) being evaluated for structured programming use
____ most often used

Some opinions expressed or facts reported are noteworthy:

1. MPL, a program design language, is being developed at JPL, Jet Propulsion Laboratory (see Volume II);

2. GSFC, Goddard Space Flight Center, is investigating effective methods of program documentation and studying systems implementation languages (see Volume II);

3. JSC, Johnson Space Center, is planning a procedure for standardization of HAL/S through the use of a central facility for compiler changes;

4. MCC, Mission Control and Computing Center (at JPL), believes that high order programming languages must have the following properties:

    a. permit retention of the large capital investment in existing software;

    b. be compatible with interactive software development;

    c. support a convenient syntax for structured programming.

5. GSFC suggests that, although in widespread use, FORTRAN not be selected as the standard high order language for NASA systems of the type which are the responsibility of Air Force Systems Command. Rather, it is suggested that language standardization may be better achieved through the development of a specific language for a given area of application, such as communications. Further, current research in language technology on structures which support data abstraction and concurrent process synchronization should be carefully analyzed for applicability to any language standardization effort.

One other phenomenon should be noted. As is the case in Air Force avionics systems, most operational flight programs for NASA applications have been written in assembly language. The notable exception is HAL/S, designed and used for manned spaceflight computer applications on the IBM AP-101. If we assume that space, weight, and timing constraints are similar to those which pertain in Air Force avionics applications, then the HAL/S experience should be further investigated to help answer the question "Can a high order language be used effectively in avionics?". For example, are the language features of HAL/S particularly well suited to the job, is the compiler especially good at optimizing, or both? The Intermetrics

report [MART75] indicates that the compiler has been carefully timed for optimization and that this has a significant impact on its usefulness. How does the use of HAL/S compare with the use of JOVIAL J3B for the B-1 and the F-16 and the possible use of J73/I in avionics applications? Are the constraints in these cases the same or different? The data in this report give no quantitative comparisons but do indicate sufficient similarity to argue that the HAL/S experience is relevant to Air Force avionics needs.

APPENDIX II

OTHER STANDARDIZATION EXPERIENCE

Experience in other countries with standard languages can provide some insights into the requirements for successful standardization as well as potential pitfalls. Since achieving high order language standardization involves more than language characteristics, the following background on CORAL 66 places in perspective many aspects of standardization. This information was assimilated at a course offered by the Navy and taught by British Ministry of Defence (MoD) personnel, supplemented by literature distributed at the course in March 1976.

CORAL 66

CORAL 66 is a high order programming language chosen by the Ministry of Defence, United Kingdom, as an interservice standard for military programming. The acronym means "Computer-On-Line Real-Time Application Language."

Background

The standardization activity which has led to the establishment of CORAL 66 as the United Kingdom's (UK) national, military programming language had its initial impetus in 1964 when a committee in the Ministry of Defence (MoD) made two major recommendations:

- do not develop new hardware for specific applications; use off-the-shelf equipment*;
- MoD should adopt a high order language.

At the same time, the Royal Radar Establishment (RRE) was working on producing tools for building compilers. Simultaneously, people working on LINESMAN (the UK air defense system) were complaining bitterly about multiplicity of programming languages and compilers. They were using CORAL 64 (an amalgam of JOVIAL and Algol) among others; they found that the language did not lend itself to easy compilation, given the compiler production tools available at the time. It was costing about $250,000 (at today's prices) for each CORAL 64 compiler and this was considered excessive.

---

*This approach worked well except for the avionics area.

The MoD policy direction, the Royal Radar Establishment (RRE) compiler-building tools production, and the LINESMAN problems conspired to foster a reappraisal, during the period from 1964 to 1966, leading to the definition of CORAL 66. The definition was made by the Royal Radar Establishment with comments by Army, Navy, Air Force, and uniformed representatives of potential users.

There was reluctance to use CORAL 66 until one or two systems had been produced, more or less in research environments. It was used, in the period from 1966 to 1970, for computer-driven radar processing and for writing compilers, operating systems, and support software. In 1970 the standard definition was published.

Language

CORAL 66 is a general-purpose programming language based on ALGOL 60, with some features from CORAL 64, JOVIAL, and FORTRAN. It was designed in 1966 by Currie and Griffiths of the Royal Radar Establishment, United Kingdom, in response to the need for a compiler on a fixed-point computer in a control environment; this was partially motivated by a large military project in which $2.5 million had been invested in the CORAL-64 language and its associated software [WOOD73, DEPL75]. The redefinition of CORAL in 1966, resulting in CORAL 66, caused the language to bear a closer relationship to ALGOL 60 than any other language.

A CORAL 66 program consists of communicators and separately compiled segments. Each segment has the form of an ALGOL 60 block, within which blocks and procedures may be nested to arbitrary depth. The purpose of a communicator is to specify and name those objects which are to be commonly accessible to all segments. One type of communicator, COMMON, allows different segments of the same program access to a common set of data. The other communicators are LIBRARY (access to library procedures and data), EXTERNAL (access to an object completely external to a CORAL 66 program), and ABSOLUTE (access to objects having absolute addresses in the computer in which a CORAL 66 program executes).

The basic structure of a CORAL 66 program is:

name of program

COMMON,EXTERNAL,LIBRARY, and/or ABSOLUTE

segment-name-1

BEGIN . . . segment 1 . . . END;

```
segment-name-2

BEGIN . . . segment 2 . . . END;

.

.

.

segment-name-n

BEGIN . . . segment n . . . END;

FINISH
```

CORAL 66 allows IF-THEN-ELSE statements and two types of FOR statement, those with STEP-UNTIL specification and those with WHILE condition specification. Thus, it is possible to write CORAL 66 programs which adhere to generally accepted practices of structured programming.

The definition of standard CORAL 66 is contained in a 58-page pamphlet published by the Ministry of Defence, United Kingdom [WOOD73]. The definition of the language has remained essentially unchanged since its original definition in 1966.

Standardization

The objectives sought in the design of the CORAL 66 language were:

- to have a language for which it would be possible to compile efficient (speed and memory) object code;
- to reduce development time of a system and improve its integrity and documentation;
- to be able to produce compilers which themselves would be efficient; and
- to minimize compiler development costs [ENSL75].

In 1970 CORAL 66 was established as a military standard by the MoD. However, the establishing directive went only so far as to say that CORAL 66 was the preferred language rather than required.

Mr. Neve, RRE, in commenting on this point at a CORAL 66 seminar given in February 1976 in Reston, Virginia, indicated that the MoD's choice of 'preferred' was a mistake; he claimed that they should have been rather more strict about it in order to avoid the initial hassling which occurred because of reluctance to use a new standard language. As of January 1973, however, it became a requirement that computers used for weapons systems have a CORAL 66 compiler [ENSL75].

Over the past six years, however, reluctance seems to have turned into something close to enthusiasm as a result of insistent support for the language and pressures to produce compilers for it. The UK MoD does not provide funds for compiler development. It has an established list of preferred computers for system acquisitions. This list has about 45 machines on it, each of which has a CORAL 66 compiler which has successfully gone through an assessment by MoD using test programs. The assessment establishes whether a compiler meets the standard and checks sufficiency with respect to other compilers. As of January 1976 the assessment capability consisted of a suite of 18 tests and 6 benchmarks. When a compiler has passed assessment the computer is added to the list. Failures are also made public, but vendors can try again. (A review of the documentation is used to check for extensions to the standard, but this does not affect passage of the assessment.) Any contractor wishing to build a CORAL 66 compiler can seek assistance through the MoD, gaining the advantage of others' experience.

In 1973 CORAL 66 became a *de facto* national standard as British industry began to use it; this was supported by the action of the Department of Industry, UK, which adopted CORAL 66 in 1973 and set up a support organization [ENSL75]. British Steel did a study and picked CORAL 66 for the principal reason that it was available, not because of language features or elegance. Another factor heavily influencing British industry is that CORAL 66 compilers are relatively inexpensive to build. An average CORAL compiler, according to Mr. Neve, takes 12-16K words, single pass. If frills are added, its size may reach 32K words. Typical cost for a CORAL compiler is $60,000.

CORAL 66 Use

CORAL 66 is now in use by all the military departments within the MoD, by some non-military government organizations in the United Kingdom, and by a significant segment of British industry. It is used primarily on small to medium scale computers for a great variety of applications. Some United States companies are now involved in CORAL 66 compiler building: DEC, PDP-11, contract with UK Air Training Center; VARIAN; IBM; and Honeywell, DDP-316/516/716.

Depledge [DEPL75] indicates compilers available or under development in 1975 as indicated in Table IX.

Table IX

CORAL 66 Compilers Available or Under Development in 1975

IBM 360/370	ITT 3200
ICL 1900	ITT 1650
ICL 2900 Range	General Automation SPC 16
ICL 2903	Computer Technology Modular One
ICL 7903	DEC PDP 11
ICL System 4	DEC PDP 9
GEC Myriad I, II, III	DEC PDP 15
GEC 900	Honeywell 316
GEC 2050	Honeywell 516
GEC 4080	Honeywell 716
GEC Mark 2B	Digico Micro 16
GEC Locus 16	Arcturus 18D
Ferranti FM 1600	Datapoint 2200
Ferranti Argus 400	Hewlett Packard
Ferranti Argus 500	National Semiconductor IMP 16
Ferranti Argus 700	XEROX 550
Sperry 1412	XEROX 560
Plessey XL4	Sigma 9
Plessey XL9	Sigma 6
Plessey System 250	Kongsberg KS 500

Real Time Faciities and I/O Capabiity

There are no CORAL 66 standard language statements for timing, interrupt handling, or I/O. Programmers access machine capabilities via machine language code which is activated either through procedure call or macro usage, as is done in other languages in the class of CORAL 66. Some argument in favor of this arrangement can be brought forward; here is a quote from an RRE publication [RRE 74]:

> Timing and interrupt facilities are not standardized in CORAL 66, as the language is intended to be suitable for a wide variety of computers with different supervisory software. The programmer's control over external events, and the computer's reaction to them, must be expressed by calls of procedures or macros with bodies designed to interface with whatever facilities are normally provided by the computer manufacturer. No fixed conventions are laid down, but the parameter mechanism for procedures and macros is sufficently powerful to permit definition of useful real-time statements at the language level.

The lack of I/O statements in CORAL 66 has caused a problem, however, insofar as there is no standard which specifies how I/O is to be implemented. Within the past year there has been an effort started to produce a draft standard for simple character string in/out. According to Mr. Neve, the British, RRE personnel in particular, consider the lack of I/O definition a standardization problem, not a language deficiency.

Summary

The CORAL 66 standardization activity in the UK overall seems to have achieved a net benefit over the past ten years. After a period of reluctance the military establishment began to use CORAL 66. This led to greater availability of CORAL 66 compilers, which in turn encouraged the use of CORAL 66. In 1973 the use of CORAL 66 spread outside the MoD into British industry. Users have generally been enthusiastic about the cost savings associated with using the established standard language; no large groups argue that the language itself is particularly outstanding, although its simplicity is sometimes seen as an advantage. So far as can be determined the important benefits gained in the UK from standardizing on CORAL 66 are:

- the inventory of compilers is large enough to provide incentive to vendors to develop compilers if they do not have one in order to compete successfully; this gives

the buyer more objectives when he designs for his application;

- the list of preferred computers maintained by the MoD provides guidance to buyers and incentive to vendors;

- because the original intention of making it easy and cheap to build compilers was followed (partly by keeping the language simple and partly by almost always disapproving proposed changes/enchancements), it is simple and inexpensive to build a CORAL 66 compiler;

- for any acquisition two sets of CORAL 66 guidelines can be given to vendors -- first, the official CORAL 66 definition, a well-written 58-page pamphlet and, second, specialized guidelines oriented toward the particular acquisition and based on previous experiences;

- personnel transportability has been achieved;

- validation of compilers is expedited: packages for syntax checking and benchmarking are available, results of tests are made public, and vendors have every opportunity to get their compilers up to grade before testing;

- an atmosphere of cooperative development of systems (cooperation between buyer and vendor) has been enhanced by the existence of and experience with the standard language;

- many compiler development tools are available, contributing to the low cost of a new compiler.

Unfortunately, the British experience can tell us nothing significant yet about what to do when the standard language becomes painfully obsolescent in the face of new technologies and more complex application requirements. Perhaps the only hint is the obvious one -- do it all over again with a new language; in that case, at least the UK will have done it once and will have the techniques and organizations to do it again more easily than the first time.

APPENDIX III

COMPUTERS

Table X lists those computers employed in the sixty-four systems reported in Volume II. Of the forty-two computers employed, twenty-two are classified as major hardware systems. These systems are further broken down into ten large-scale processors, and twelve medium to small-scale computers. The twenty remaining computers are listed as micro and flight computers, and find one-time use in the reported systems. Short descriptions including word size, memory size, and languages supported, are given for the major hardware systems. The smallest unit of memory size is the bit. For each computer the number of bits of data which make up a byte or word are given. Available memory sizes are described in Kilobytes or Kilowords; abbreviate K, a kilobyte equals 1024 bytes and a Kiloword equals 1024 words. In some cases, memory size is given in Megabytes (1024 Kilobytes) or Megawords (1024 Kilowords), abbreviated M.

Table X

Hardware Listing

Large-Scale Computers

1. Burroughs 700 Series/D Machine
2. CDC 5600 (1700), AN/UYK-25
3. CDC 6000/7000 Series, CYBER 70 Series
4. DEC System 10
5. Honeywell 6000 Series
6. Hughes 118 Series
7. IBM 360/370
8. IBM 7090
9. UNIVAC 1100, AN/UYK-7
10. UNIVAC Series 70

Medium-Scale Computers and Minicomputers

11. UNIVAC 1600, AN/UYK-20 (AN/UYK-15)
12. CDC 3000 Series
13. Data General Nova, Rolm 1602 (AN/UYK-19), Rolm 1603 (AN/UYK-27)
14. DEC PDP-8
15. DEC PDP-11
16. DEC PDP-15
17. Harris S-120
18. Honeywell 16 Series/Datanet 355
19. IBM system/4Pi
20. MODCOMP
21. Raytheon RDS-500
22. Xerox 550, Sigma Series

Micro and Flight Computers

1. Adage - AGT 50
2. AN/FYK-5
3. Autonetics PPS-4
4. Computeck 200
5. D-37D airborne computer
6. Datacraft 6024
7. Delco M362-F
8. HP 2100, 21MX
9. HP 9500
10. Hughes 81
11. Intel 8080
12. Interdata 770
13. Lear Siegler
14. Northrup NDC-1051A
15. RCA SCP-234
16. SEL-32/55
17. Singer SKC 3000
18. TI 980, 1093, 2520-2, and 2540
19. UNIVAC 1230
20. Westinghouse modified milli
21. AN/AYK-15

HARDWARE DESCRIPTIONS

Burroughs Corporation 700 Series/D Machine (1)

Burroughs 700

The Burroughs 700 Series are small-scale business minicomputers. Basic machine functions are stored in separate read-only memories. Logic conversions are done by interpreter programs. High priority programs may be run on an "interrupt/resume" basis in place of multiprogramming. Memory capacities range from 16 to 100K 64-bit words. Languages supported include COBOL and RPG.

Burroughs D Machine

The Burroughs "D machine", a specialized product from the 700 Series, is a medium-scale military communications computer capable of interactive and batch processing. The "D machine" is microprogrammable and operates under the Supervisory Control Program (SCP) which is primarily a serial batch system. Languages available include assembly, COBOL, RPG, and the Network Definition Language (NDL). Storage on the "D machine" consists of a core or semiconductor memory with a capacity of 64K 16-bit words.

Control Data Corporation (CDC) 5600 (1700), AN/UYK-25 (2).

The CDC 5600 and the similar AN/UYK-25 are small general-purpose microprogrammable digital computers with word size ranging from 8 to 32 bits in 4-bit increments. In the applications reported, it is used to emulate the CDC 1700.

The CDC 1700 operates under the Mass Storage Operating System (MSOS). Languages supported include BASIC, FORTRAN, and assembly.

Core memory capacity ranges from 4K 8-bit words to 262K 32-bit words.

Control Data Corp. (CDC) 6000/7000 Series, CYBER 70 Series (3)

Control Data's CYBER 70 is a series of four large-scale general purpose computers. Models 72, 73, 74 (large) offer multiprocessing while the model 76 (very large) does not. The CYBER 70 is based entirely on the CDC 6000 series architecture.

The CYBER operates under two systems, the NOS system which offers the BASIC, FORTRAN, COBOL, and APL programming languages, and the SCOPE batch-processing system which offers ALGOL, BASIC, COBOL,

COMPASS (assembly), FORTRAN IV, and JOVIAL. PL/I, PROSE, RPG, and SNOBOL are also available.

The CYBER offers a maximum core memory of 131K 60-bit words.

Digital Equipment Corporation (DEC) System 10 (4)

The DEC System 10 is a series of medium to large-scale general-purpose computers capable of batch, time-sharing, real-time, and dual processing operations. All six models offer multiprogramming.

System 10 is operated by the TOPS-10 Operating System which permits concurrent operation of time-sharing, batch, real-time and remote configurations.

Memory capacity ranges from 64K to 4096K 36-bit words. Languages supported include COBOL, FORTRAN, ALGOL, BASIC, MACRO-10 assembly, and APL.

Honeywell 6000 Series (5)

Honeywell's 6000 series consists of six single-processor models that feature multiprogramming as a standard mode (the four largest models also offer multiprocessing). Models 6030, 6050, and 6070 are scientific/engineering oriented. Models 6040, 6060, and 6080 are business oriented.

The 6000 series is operated under the General Comprehensive Operating Supervisor (GCOS). Core memory capacity varies between models; maximum storage ranges from 262K 36-bit words on the 6030 to 1M 36-bit words on the 6080. Languages supported include FORTRAN, COBOL, BASIC, ALGOL, JOVIAL, and ABACUS.

Hughes Aircraft 118 Series (6)

The Hughes Aircraft H-5118M is part of the 118 series of military computers which also include the H-4118 and H-3118 models.

The H-5118M offers semiconductor memory with a capacity of 124K 18-bit words. Languages available include JOVIAL and HAP assembly.

International Business Machines System 360/370 (7)

IBM's medium to large scale computer system consists of 19 central processor models designed to handle a broad range of environments. System 370 is available with virtual storage and multiprocessing capabilities.

Principal 360/370 operating systems are the Basic Control System (BCS), the Disc Operating System (DOS), and the IBM Operating System (OS). Two versions of OS, the OS/MFT and OS/MVT, offer multiprogramming with a fixed number of tasks and variable number of tasks, respectively. The OS/VS supports virtual storage.

Bipolar memory is offered with capacity ranging from 131K to 1M 32-bit words. Languages supported include FORTRAN, BASIC, RPG, COBOL, PL/I, APL, ALGOL, and assembly.

International Business Machines System 7090 (8)

IBM's System 7090 is a large scientific computer designed primarily for solving complex mathematical problems.

The 7090 operates under the IBSYS Operating System. Languages supported include FAP and MAP assembly and FORTRAN.

Memory has a capacity of 32K 36-bit words.

UNIVAC 1100 Series, AN/UYK-7 (9)

The UNIVAC 1100 Series is a line of medium large to very large general-purpose computers. Models include the 1106, 1108, 1110, 1100/20, and the 1100/40, and are available as either single or double processors.

Series 1100 operate under the EXECUTIVE Operating System. Core or semiconductor memory is available which ranges from 32K to 1M 36-bit words. Languages supported include COBOL, FORTRAN, SIMSCRIPT, BASIC, JOVIAL, PL/I, ALGOL, APL, NUALGOL, RPG, and assembly.

AN/UYK-7

The AN/UYK-7 was developed by UNIVAC as a general-purpose, 32-bit third generation computer featuring core memory modules of 48K words each, expandable to 260K.

UNIVAC Series 70 (10)

The UNIVAC Series 70 (formerly RCA Spectra 70) is a line of general-purpose virtual memory machines with multiprogramming capabilities. Series 70 is comprised of ten models which are completely compatible with the IBM 360/370 systems.

Principal Series 70 operating systems include DOS, TDOS, and VMOS. Core memory capacity ranges from 65K to 524K 32-bit words.

Languages supported include COBOL, FORTRAN, RPG, BASIC, ALGOL, and assembly.

UNIVAC 1600 Series, AN/UYK-20 (AN/UYK-15) (11)

The 1600 Series are flexible 16-bit machines with direct interfaces to other Series 70 processors. The 1600 is well suited to handling communications front-end requirements.

Memory capacity ranges from 8K to 65K bytes. Languages supported include COBOL, FORTRAN, RPG, and assembly.

AN/UYK-20

This is a UNIVAC general purpose digital computer capable of a variety of tactical applications requiring moderately small amounts of processing.

The AN/UYK-20 is a 16-bit machine with memory-module capacity of 48K words, expandable to 260K. The computer operates under the Compiler Monitor System - 2nd generation (CMS-2) which includes the high level tactical programming language.

Control Data Corporation (CDC) 3000 Series (12)

The CDC 3000 Series is a line of medium-scale general-purpose computers. Current models include the 3100, 3170, 3300, and 3500. The 24-bit 3200 model and the 48-bit 3800 model are no longer marketed.

The 3000 Series are operated by the dual processing MASTER Operating System and the Mass Storage Operating System (MSOS). Memory capacity ranges from 8K to 262K 24-bit words. Languages supported include COBOL, ALGOL, BASIC, FORTRAN, and COMPASS (assembly).

Data General Nova and Supernova Series, Rolm 1602 (AN/UYK-19), Rolm 1603 (AN/UYK-27) (13)

The Nova/Supernova are small-scale general-purpose computers, oriented toward control, scientific, laboratory, and time-sharing applications.

All models operate under the DOS and RDOS disc operating systems. Memory capacity ranges from 256K to 768K 16-bit words. Languages supported include FORTRAN IV and V, BASIC, ALGOL, and assembly.

Rolm 1602 (AN/UYK-19)

The Rolm 1602 (AN/UYK-19) is a ruggedized computer designed for severe environment applications. It features an interrupt structure, an expanded instruction set, extensive I/O interfaces and upward compatibility with the Nova series. Memory capacity ranges from 8-164K 16-bit words. Languages supported include ALGOL, BASIC, and FORTRAN.

Rolm 1603 (AN/UYK-27)

The Rolm 1603 (AN/UYK-23) is a ruggedized minicomputer for military and severe environment applications. It includes over forty peripherals and interfaces. Memory capacity ranges from 8-32K 16-bit words. Languages supported include ALGOL, BASIC, and FORTRAN.

Digital Equipment Corporation (DEC) PDP-8 (14)

The PDP-8 is a line of three core-based minicomputers utilizing the same basic instructions and processing operations. Current PDP-8 mainframe architecture is centered around the OMNIBUS which carries control, timing, and data signals connecting all major systems.

The major operating system for the PDP-8 is the OS/8 which supports both batch and interactive processing. Languages supported include FORTRAN IV, BASIC, ALGOL, DIBOL, FOCAL, and assembly.

The PDP-8 memory consists of one to eight memory modules each providing a capacity of 4K 12-bit words. Memory capacity ranges from 4 to 32K 12-bit words.

Digital Equipment Corp. (DEC) PDP-11 (15)

Digital's PDP-11 is a series of ten micro to midi size computers. Models include the 11/03 (micro), the 11/04, 11/05, 11/10, 11/34, 11/35, 11/40, 11/45, and 11/55 (minis), and the 11/70 (midi).

All PDP-11 models, except the 11/45, are organized under a single fast UNIBUS that connects all system components. Core, MOS, or bipolar memory are available with capacities ranging from 8K-128K 16-bit words. Operating systems include a Paper Tape Software System and a Cassette Programming System (CAPS), Resource Time-Sharing System (RTST), Disc Operating System (DOS), and Real-Time Multiprogramming Systems (RSX-11D, M, and S). Current languages supported include the PAL-11 and MACRO assembly languages, FORTRAN IV, FOCAL, BASIC, COBOL, ALGOL, and MUMPS-11.

Digital Equipment Corporation (DEC) PDP-15 (16)

The PDP-15 is a powerful 18-bit minicomputer system designed for laboratory, control, scientific, and mathematical applications.

Operating systems for PDP-15 include FORTRAN-oriented DOS-15 and BUS-15 systems, Advanced Software System (ADSS), and RSX PLUS III. Languages supported include ALGOL, FORTRAN IV, and assembly.

Main storage is a core memory ranging from 4K to 132K words.

Harris Corp. S-120 (17)

The Harris Series 100 is a 24-bit minicomputer line, designed for high speed, real-time, scientific applications.

All Slash series models operate under the VULCAN (Virtual Core Manager) system. Core and semiconductor memory is available with capacity ranging from 8K to 262K words. Languages supported include ALGOL, BASIC, FORTRAN, COBOL, RPG II, and SNOBOL.

Honeywell 716 (16 Series), Datanet 355 (18)

Honeywell 716

The 716 processor is a 16-bit, word-oriented, single address system. All 16 Series systems can operate under OS/700 Disc or Core Operating Systems (DOS or COS) which are multiprogramming, real-time operating system. Languages supported include FORTRAN IV, BASIC and assembly.

Main storage on the 716 consists of one to eight 8K word memory modules.

Datanet 355

The Datanet 355 is a programmable front-end network processor which controls remote terminals connected to Honeywell Series 6000, 7000, 600, 200 and 100 and Levels 66 and 68 computer systems. It is designed for large-volume communications applications.

IBM System/4Pi and Advanced System/4Pi (19)

System/4Pi is a series of general-purpose, military digital computers designed to handle a wide range of real-time processing and multiprocessing requirements. System/4Pi computers now include three

types: the advanced processor (AP) series, the command and control (Model CC), and the subsystem processor (SP) machine.

Core memory capacity range from 64K to 176K 32-bit words. Languages supported include JOVIAL (via cross-compiler) and assembly.

Modular Computer Systems (MODCOMP) (20)

The MODCOMP systems are a series of four highly modular, microprogrammed 16-bit machines. Each of the models (I-IV) are designed for general applications. Model IV offers dual processing.

MODCOMP operates under a variety of real-time and batch operating systems including MAX, MAXNET, and MAXCOM. Core memory ranges are: MODCOMP I - 2K to 32K 16 bit words, MODCOMP II - 8K to 64K 16-bit words, and MODCOMP IV - 16K to 256K 32-bit words. Languages supported include FORTRAN IV, BASIC, and assembly.

Raytheon RDS-500 (21)

The RDS-500 is a general purpose minicomputer capable of high speed, real-time processing such as aircraft simulation control and closed loop process control.

The RDS-500 operates under the Standard Operating System (SOS), the Magnetic Tape Operating System (MTOS), and the Real-Time Operating System (RTOS). Main storage capacity ranges from 8K to 66K 16-bit words. Languages supported include FORTRAN, RPGII, and assembly.

Xerox 530, 550, and 560, Sigma Series (22)

The Xerox 550 and 560 are two new medium-scale computer systems oriented to real-time scientific applications. Capabilities range from general purpose processing on the 530 to multiprocessing on the 560.

Xerox operates under the CP-R (real time) and the CP-V (time-share) operating systems. Core memory capacity ranges from 8K to 262K 32-bit words. Languages supported include FORTRAN, FORTRAN IV, APL, BASIC, RPG, COBOL, and assembly.

Xerox Sigma Series

The Xerox Sigma Series (now replaced by the 550 and 560 models) is a line of medium to large-scale computers oriented to real-time

and scientific applications. Models include the Sigma 3, 5, 6, 7, 8, and 9.

Operating systems for the Sigma Series include the Basic Control Monitor (BCM), Real Time Batch Monitor (RBM), Batch Time-Sharing Monitor (BTM), Control Program for Real-Time (CP-R), and the Control Program-Five (CP-V).

Memory capacity ranges from 8K to 524K words. All models offer 32-bit words, except model 3 which offers 16-bit words. Languages supported include BASIC, FORTRAN, APL, and RPG.

APPENDIX IV

EXPERIENCE WITH SOFTWARE ENGINEERING TECHNIQUES

The Air Force systems and NASA centers below have used some form of software engineering, most notably structured programming. See Volume II for greater detail.

COMBAT GRANDE

Top-down structured design was employed with module sizes of approximately 100 instructions per module.

E-3A (AWACS)

Assembly language routines for four TDMA were developed by Hughes using structured programming techniques, most notably HIPO and chief programmer teams.

SATIN IV

Top-down structured programming was required for software development. Correctness of trusted modules will be verified.

TRI-TAC/TCCF

DSPL (Display Processing Language) was designed to facilitate structured programming.

CONUS OTH

TRW developed application software using some structured programming techniques although these techniques were not required.

JTIDS/ASIT

Structured programming is required; sections of [RADC75] are specifically required.

JSS

Specifications suggest use of certain structured programming techniques, but their use is not a formal requirement.

PAVE PAWS

Specification required use of top-down structured programming as specified in certain subsections of [RADC75]; additional volumes of [RADC75] were listed as guidelines.

TRACALS PIDP

Some bidders are proposing use of structured programming techniques, including top-down design, indentation and comments; no new language (i.e., assembly language) constructs are being developed or used.

ASTROS

This is a structured programming feasibility study being performed at SAMTEC using structured walk-throughs, top-down design, HIPO, program support library, chief programmer teams, and structured coding supported by S-FORTRAN, a FORTRAN processor.

RISS

FORTRAN support packages have assisted user software development, debugging, and updating.

NASA

Several centers are using software engineering techniques.

G. C. Marshall Space Flight Center (MSFC)

MSFC currently uses only FORTRAN and COBOL to reduce costs. This restriction is expected to be relaxed soon to allow use of a language which lends itself to structured programming; FORTRAN and APL are being considered. At present, a FORTRAN preprocessor is being used experimentally for writing structured programs.

Johnson Space Center (JSC)

HAL/S was "designed for structured programming and other advanced techniques."

Goddard Space Flight Center (GSFC)

Goddard Space Flight Center (GSFC) has had one set of application programs written using structured programming techniques; this was done by IBM using assembly language for TELOPS (Telemetry On-Line Processing System). GSFC has reported that structured programming techniques are being used to an increasing extent. A set of macroinstructions has been developed by the Science and Applications Computing Center (SACC) of GSFC to supplement assembly language on IBM 360 computers for structured programming. A structured programming FORTRAN preprocessor is also being implemented by the SACC.

Jet Propulsion Laboratory (JPL)

Jet Propulsion Laboratory has reported that top-down, structured programming will be used for the Deep Space Network. SFTRAN (a FORTRAN preprocessor for structured programming) is being used at JPL's Mission Control and Computing Center (MCCC). MCCC has had favorable reaction from its programmers to the introduction of structured programming technology. In 360 assembly language and FORTRAN, some experience has been gained with the use of preprocessors or macros which permit the use of structured programming syntax; improved productivity has been noted in this experience.

APPENDIX V

GLOSSARY OF SYSTEMS

The following glossary contains descriptions of each of the sixty-four Air Force systems, the NASA centers, and other government centers included in Volume II of the report and summarized in Volume I. These descriptions are similar to the overview and narrative description which accompanies each system or center in Volume II.

ACTS

Automated Communications Test Software for Fleet Satellite Communications (FLTSATCOM)

A software system used by the FLTSATCOM prime contractor to control automated test equipment. It simulates communication ground station functions and performs extensive system level performance tests. A combination of the software and hardware enables testing of all communication system level performance parameters of the FLTSATCOM communications satellite program.

ADTC

Armament Development and Test Center, Eglin Air Force Base, Florida

An organization which manages the Air Force non-nuclear munitions program. It also conducts research and development testing of aeronautical systems such as aircraft and their associated missiles and airborne electronic warfare devices.

AFAL

Air Force Avionics Laboratory, Wright-Patterson Air Force Base, Ohio

The laboratory conducts research and technology programs for Air Force electronic components, optics and photo materials, navigation and guidance, vehicle defense, electronic warfare and communications.

AFEES

Automated Armed Forces Examining and Entrance Station

This program entails the design, development, test and evaluation of a prototype automated AFEES that will substantially improve examinee screening and administrative processing within the AFEES.

AFSATCOM I

Air Force Satellite Communications I

The program is for the acquisition of ultra high frequency (UHF) airborne/ground force terminals, airborne/ground command post

terminals, ancillary equipment necessary for operational control and communications transponders on selected Air Force satellites. The associated family of modular UHF transceivers will provide a command communications capability in the line-of-sight (LOS) mode.

AFSATCOM II/III

Air Force Satellite Communications II/III

A system providing reliable and secure means for complete command and control of weapon systems during crises. It provides the ability to communicate with globally dispersed forces.

AFSCF

Air Force Satellite Control Facility, Los Angeles Air Force Station, California

A worldwide on-orbit tracking and control network for Department of Defense (DoD) space programs. It is headquartered at Los Angeles Air Force Station, California. Satellite data are collected and processed through a combined network/mission control center, the Satellite Test Center (STC), remote tracking telemetry and command stations, a radiometric test facility and a space recovery organization.

ASTROS

Advanced System Techniques for Reliable Operational Software

A Space and Missile Test Center/Rome Air Development Center (SAMTEC/RADC) project to validate productivity claims of various software vendors and to establish a data base of statistics gathered in a military operational environment. ASTROS concentrates on the investigation and validation of structured programming, measurements of the benefits derived by applying those techniques and the objective evaluation of data gathered.

ATEC

Automated Technical Control

A computer-assisted capability for Defense Communications System (DCS) technical control facilities. The computer-assisted portion of the overall Technical Control Improvement Program supplements equipments presently installed under the manual grade portion of the improvement program.

AWACS

Airborne Warning and Control System

See E-3A.

B-1

Strategic Bomber (Rockwell International)

A blended wing-body configuration aircraft to provide modernization of the strategic bomber force. It is designed to cruise at subsonic speeds and attack at high subsonic speeds at low altitude, or in an over-the-target supersonic dash at high altitude.

C-5

Galaxy, Cargo Transport Aircraft (Lockheed)

A very heavy logistics transport aircraft of the Military Airlift Command (MAC). It is currently the largest aircraft in service anywhere in the world. C-5 avionics include two computers, one used in the back-up mode. These provide the aircraft with comprehensive navigation capabilities and built-in test functions.

CCPDS

Command Center Processing and Display System

A near-real-time computerized operational display system which can assimilate and display Strategic Air Command Automated Command Control System (SACCS) data for Commander-in-Chief Strategic Air Command (CINCSAC). It receives satellite sensor warning data

concerning missile lift-off from an external network. These data are correlated with selected elements of Stratetic Air Command (SAC) mission data to satisfy critical CINCSAC requirements in command and control of SAC forces. Analyses of the warning data are displayed in the SAC Command Post and sent to the E-4 via SATIN IV. The CCPDS is also known within SAC as the SAC Warning and Control System-Offutt Subnet Communications Processor (SWCS-OSCP).

COBRA DANE

Phased Array Radar, Shemya Island, Alaska

The acquisiton effort for a phased array to be installed on Shemya Island, Alaska. The system collects and disseminates intelligence data on Soviet ballistic missile test firings, detects and warns of missile firings impacting on the Continental United States (CONUS) and collects and disseminates data on earth satellite vehicles.

COMBAT GRANDE

Semiautomated Spanish Air Defense System

Upgrade, modernization, and semiautomation of the existing Spanish Air Force aircraft control and warning (AC&W) network.

CONUS OTH

Continental United States Over-the-Horizon Radar System

It provides long range detection of aircraft approaching North America. The Over-the-Horizon-Backscatter (OTH-B) radars will be part of the North American Air Defense Command (NORAD) system that provides an air surveillance and warning capability. The distinguishing technical feature of the OTH-B is its capability to detect targets at all altitudes and extended ranges. The present phase of this program is to build a prototype OTH-B radar, test it for a year and then make a decision on building two fully operational radars.

CSDRO

Computer Services Division Range Operations, Vandenberg Air Force Base, California

A missile range operations capability at Vandenberg Air Force Base, California.

DFRC

Dryden Flight Research Center, Edwards Air Force Base, California

A National Aeronautics and Space Administration (NASA) facility concerned with manned flight within and outside the atmosphere, including low speed, supersonic, hypersonic and reentry flight, and aircraft operations.

DMSP

Command and Control Support - Defense Meterological Satellite Program

An advanced satellite system which provides imagery and other specialized meterological data in support of specialized strategic and tactical operations. Two polar orbiting satellites provide data directly to major decision making points and global coverage to the Air Force Global Weather Central (AFGWC). The AFGWC disseminates selected data to key command and control points via the Digital Data Facsimile System. The primary function of the DMSP Command and Control Center (CCC) is on-orbit operational control of all DMSP satellites. It accomplishes this by an Integrated Commanding System (ICS) for the control and monitoring of command load data transmitted to each Command Readout Station (CRS).

DMSP

Ground Segment-Defense Meterological Satellite Program

An advanced weather satellite system which provides imagery and other specialized meterological data in support of specialized strategic and tactical operations. The DMSP Ground Segment includes: Command Readout Stations (CRS) for real-time command and control of satellites collection of data from them and data relay to the Air Force Global Weather Central (AFGWC), Data Reconstruction Stations (DRS) to reconstruct and process data transmitted real-time

and post-pass from each CRS, a Command and Control Center (CCC) where satellites are commanded and controlled and receive stored data read from recorders onboard spacecraft and a Payload Test Facility (PTF) for system checkout at launch and on-orbit analysis.

DMSP

Space Segment-Defense Meterological Satellite Program

An advanced weather satellite system which provides imagery and other specialized meterological data in support of special strategic and tactical operations.DMSP space segment satellites in sun-synchonous polar orbits continuously collect global weather data in the visible and infrared spectra. Final data products are either in computer program format or film product directly usable for imagery analysis.

DS&A

Data Services and Analysis Program

An Aerospace Ballistic Recovery Entry System (ABRES) program for research and development scientific data processing performed by a Space and Missiles Systems Organization (SAMSO) contractor. The contractor is furnished ABRES computer time as Government Furnished Equipment (GFE).

E-3A

Airborne Warning and Control System (AWACS)

This system provides a survivable airborne (Boeing 707) air surveillance capability and command, control and communications functions. Its distinguishing technical feature is the capability to detect and track aircraft operating at high and low altitudes over both land and water. It will be deployed by the Tactical Air Command (TAC) in both initial phases of hostilities and in protracted situations. For the Aerospace Defense Command (ADC), it provides an efficient solution to the requirement for survivable strategic air defense surveillance and control.

E-4 Block I

AABNCPI Advanced Airborne Command Post

The system provides the National Command Authority (NCA) and the Strategic Air Command (SAC) with an improved communications, command and control system. The system will utilize some combination of automatic data processing and peripheral equipment accessed through remote terminals installed in a large wide-bodied jet aircraft (Boeing 747) that will be operable during the pre-, trans-, and post-attack phase of a general war.

E-4 Block II

AABNCPII Advanced Airborne Command Post

A Boeing 747 aircraft equipped with advanced Command Control Communications (C3) equipment. It is to serve as the National Emergency Airborne Command Post (NEACP) and the Hq. Strategic Air Command Airborne Command Post (Hq. SAC ACP).

EF-111A

Tactical Jamming System (Vought)

An Electronic Countermeasures (ECM) version of the F-111A tactical fighter with improved engine performance. It is capable of locating enemy radars and directing Wild Weasel fighters to attack them. The EF-111A contains four operational flight computers, each with unique computation control and integration functions.

F-15

Eagle, Air Superiority Fighter (McDonnell)

A single seat, fixed wing all weather fighter designed specifically for an air superiority role. It also has an air-to-surface attack capability.

F-16

Multipurpose Tactical Fighter (General Dynamics)

A high performance extremely maneuverable multipurpose fighter. It exploits emerging advanced technologies.

FSS

Flight Service Station

A Federal Aviation Administration (FAA) project to automate FAA Flight Service Station (FSS) capabilities. Automated FSS capabilities are to interface with Air Route Traffic Control Centers (ARTCC) of the National Airspace System (NAS).

GEODSS

Ground Based Electro-Optical Deep Space Surveillance System

A worldwide network of electro-optically instrumented scanning and tracking telescope sites. It provides surveillance and tracking capabilities of deep space objects at altitudes greater than 3000 nautical miles. GEODSS provides an interim system to expand SPACE TRACK range, coverage, accuracy and timeliness of earth orbit satellite surveillance. This summary reports a preliminary study prior to acquisition of the worldwide network.

GERTS

General Electric Radio Tracking System, Vandenberg Air Force Base, California

A currently operational guidance and radar tracking system for expendable launch vehicles at Vandenberg Air Force Base, California. The present system has been operational for seven years. It is undergoing system reliability upgrade by changing the GERTS computer and minimizing software changes.

GSFC

Goddard Space Flight Center, Greenbelt, Maryland

A National Aeronautics and Space Administration (NASA) flight center responsible for a broad variety of unmanned earth-orbiting satellite and ground-rocket projects. It is the nerve center for the worldwide tracking and communications network for both manned and unmanned satellites.

IDHS

Intelligence Data Handling System

Communications devices and services required for intelligence communications among computer sites and between computers and remote users.

JSC

Lyndon B. Johnson Space Center, Houston, Texas

A National Aeronautics and Space Administration (NASA) center which designs, tests and develops manned spacecraft and selects and trains astronauts. It directs the Space Shuttle program. Mission Control for manned spaceflight is located here.

JSS

Joint Surveillance System

A combined USAF/Canadian program to provide peacetime air surveillance of North America. Joint FAA/USAF sites are used to fulfill the civil mission of air route traffic control and the military mission of continental air sovereignty. The Royal Canadian Air Force (RCAF) and Ministry of Transport support the Canadian portion. JSS replaces the Semiautomatic Ground Environment/Backup Intercept Capability (SAGE/BUIC) existing peacetime system. In time of crisis it provides rapid transition capability of command control and surveillance functions to the Airborne Warning and Control System (AWACS).

JPL

Jet Propulsion Laboratory, Pasadena, California

The Jet Propulsion Laboratory (JPL), Pasadena, California is operated for the National Aeronautics and Space Administration (NASA) by the California Institute of Technology (CIT). Its primary role is investigation of the planets. It also designs and operates the Deep Space Network (DSN), Mariner Jupiter/Saturn 1977 and the Mission Control and Computing Center (MCC).

JTIDS/ASIT

Joint Tactical Information Distribution System/Adaptable Surface Interface Terminal

A terminal interface for the Joint Tactical Information Distribution System (JTIDS). JTIDS is a secure, jam resistant, low intercept potential, high capacity, digital information distribution system, with a relative position/navigation capability. It interconnects the tactical forces of all services. JTIDS provides interoperability among data collection elements, combat elements and command and control centers within a military theater of operations.

KSC

John F. Kennedy Space Center, Florida

A facility which performs preflight test and prepares and launches manned and unmanned space vehicles for the National Aeronautics and Space Administration (NASA).

LORAN AN/ARN-10(V)

Tactical Long Range Radio Navigation

A program for the development and acquisition of a Navigation/Weapon Delivery System for the RF-4C and F-4E aircraft. The system provides a modular digital avionics capability with LORAN to satisfy tactical requirements for the 1978-1988 time frame.

LORAN C/D Ground Chain

Tactical Long Range Radio Navigation - AN/TRN-38(V)

A program for worldwide deployment to provide the LORAN environment for joint service common grid positioning in the tactical theater. It develops the required grid prediction and grid data management capability for joint service LORAN use.

LRC

Lewis Research Center, Cleveland, Ohio

Major programs of the Center are aircraft and rocket propulsion and electric power generation in space. Studies include metallurgy, fuels, lubricants, magnetohydrodynamics and ion propulsion. It is responsible for technical management of Agena and Centaur rocket stages.

MACIMS

Military Airlift Command Integrated Management System

An integrated, real-time, data processing system to support the Military Airlift Command (MAC) in accomplishment of its mission as the Single Manager Operating Agency for Global Airlift Services.

MINUTEMAN III

WS-1334-M and WS133B Weapon System

A three-stage, solid propellant second generation missile designed to supersede earlier intercontinental ballistic missiles (ICBM). It is an operational version which increases the

possibilities of penetrating enemy defense systems. Minuteman III incorporates an improved third stage engine and a multiple independently targetable reentry vehicle (MIRV).

MSFC

George C. Marshall Flight Center, Alabama

Launch vehicles for Apollo and other major missions are designed and developed here. The Center is concerned with launch vehicles of the Saturn class, as well as payloads, related research and studies of advanced space transportation. It is also responsible for the development of Skylab components.

NAVSTAR GPS

Global Positioning System

A joint services multisatellite navigation system. Global users are provided with extremely accurate position and velocity navigation information.

NORAD CMC Improvements

North American Air Defense Cheyenne Mountain Complex Improvement

A program to acquire new data processing equipment, software displays and communications for the North American Air Defense (NORAD) Cheyenne Mountain Complex (CMC). The NORAD Computer System, Space Information Center (SCC), and the Communications System (CS) will provide the NORAD CMC with an integrated responsive capability and a growth potential that will meet a projected life span of ten years without replacement of major equipment or major software changes.

PACOM C4

Pacific Command Command, Control, Computer, Communications

A system engineering planning effort to enhance, upgrade, and modernize Pacific Command (PACOM) capabilities.

PAVE PAWS

A system comprised of two dual-faced phased array radars, one to be deployed at Otis Air Force Base, Massachusetts and the other at Beale Air Force Base, California. The system will be operated by the Aerospace Defense Command (ADC) and will provide a warning to the National Command Authorities (NCA) of a sea launched ballistic missile (SLBM) attack against the Continental United States (CONUS).

PELSS

Precision Emitter Location Strike System

A system which accurately locates and strikes threat emitters and non-radiating targets. It provides an integrated target location of strike system capable of near-real-time detection identifications and location of emitters. It also provides precision guidance for standoff weapons to strike targets in all weather conditions.

RISS

Reconnaissance Intelligence Support System

An intelligence ground support system for the SR-71 Blackbird aircraft. It processes battlefield and multiple-sensor specialized surveillance data.

RTF

Remote Terminal Facility

A facility which interfaces with third generation dual processor replacement computers for the Strategic Air Command Automated Command Control System - Data Processing Centrals (SACCS-DPC).

SACCS-DTF

Strategic Air Command Automated Command Control System - Data Transmission Subsystem

A Strateic Air Command Automated Command Control System (SACCS) equipment subsystem which provides the communications lines, message switching and transmission equipment for the entire system. It ties Data Processing Centrals (DPC), Data Display Centrals (DDC) and the bases having SACCS input and output equipment into one large network.

SACCS/FMIS

Strategic Air Command Automated Command Control System/Force Management Information System

A generalized data management system used as the primary computer software of the Strategic Air Command Automated Command Control System - Data Processing Subsystem (SACCS-DPS). The DPS provides a capability to process and store data and to generate information for display.

SACOPS

Strategic Air Command Operational Planning System

A Strategic Air Command Automated Command Control System (SACCS) function which supports the Joint Strategic Planning Staff (JSPS) in the development of the Single Intergrated Operational Plan (SIOP). The current Data Processing Subsystem (DPS) consists of two Data Processing Centrals (DPC) at Hq. SAC.

SATIN I

SACCS-AUTODIN TTY Interface

A program to provide the Strategic Air Command (SAC) with an integrated command-wide digital communications system which will satisfy, with updating, SAC requirements for command-control, administrative and support data transmission into the 1980s. The Strategic Air Command Automated Command Control System - Automatic digital Network Teletype Interface (SATIN I) provides direct line traffic capabilities between the Joint Chiefs of Staff (JCS) and SAC units without manual relay.

SATIN IV

Strategic Air Command Automated Total Information Network

A program to provide the Strategic Air Command (SAC) with an integrated command-wide digital communications system which will satisfy, with updating, SAC requirements for command and control and which will support data transmission into the 1990s. It will be a subsystem of the Worldwide Military Command and Control System (WWMCCS). SATIN IV will provide survivable, secure, direct two-way connectivity from the communications links of the National Command Authorities (NCA) to the ground SAC combat crew commanders and the CINCSAC Command Centers. It will replace the Data Transmission Subsystem (DTS) of the Strategic Air Command Automated Command Control System (SACCS).

SDS

Satellite Data System

A multipurpose communication satellite program which in conjunction with the Fleet Satellite Communication (FLTSATCOM) satellites provides the global coverage required for the Air Force Satellite Communication System (AFSATCOM). It contains a software system developed to operate within the Air Force Satellite Control Facility (AFSCF) environment for command and control of Satellite Data System (SDS) vehicles.

SK Satellite Control Systems

Development and maintenance of program specific software for control of SK satellite systems. The software is produced for use on the computer system of the Air Force Satellite Control Facility (AFSCF) at Sunnyvale Air Force Station, California.

STEM

System Trainer and Exercise Module

A deployable trainer and exercise module of the Tactical Air Control System Improvements (TACSI) program. It provides video, simulating aircraft tracks, and audio, simulating air/ground/air (A/G/A) communications. The System Trainer and Exercise Module (STEM) furnishes the Air Force with the capability to prepare, conduct and evaluate Tactical Air Control System (TACS) Control and Reporting Center/Control and Reporting Post (CRC/CRP) training exercises.

TACC AUTO/TACSI

Tactical Air Control Center Automation/Tactical Air Control System Improvements

It provides evolutionary improvements of equipment and capabilities of communication and electronic systems for command and control of tactical aerospace operations. The system consists of automated and miniaturized equipment compatible with existing Tactical Air Control System (TACS) equipment and interfaces with automated tactical data systems of the Army, Navy and Marine Corps, providing interoperability of joint forces.

TACFIRE

Automated Field Artillery System

An Army system which automates field artillery functions through computer optimization.

TACS/TADS

Tactical Air Command System/Tactical Air Defense System

A Joint Chiefs of Staff (JCS) test program to demonstrate the secure exchange of tracking and air defense information among the tactical Aircraft Control and Warning (AC&W) systems of the services.

TFWC

USAF Tactical Fighter Weapon Center

A facility at Nellis Air Force Base, Nevada which tests and evaluates air tactics and Air Force equipment.

TIPI

Tactical Information Processing and Interpretation

The Tactical Information Processing and Interpretation/Marine Air Ground Intelligence System (TIPI/MAGIS) consists of four major segments capable of deployment at various echelons of command of the Air Force and the Marine Corps. Its purpose is to provide, through automated aids, more timely and accurate intelligence to the tactical commander. The segments include a reconnaissance photo processing segment, a photo interpretation segment, an intelligence data storage/analysis segment and an electronic intelligence processing segment.

TOSS

Terminal Oriented Support System

A Strategic Air Command Automated Total Information Network (SATIN IV) interface which provides the E-4 with the capability to communicate with the Strategic Air Command Automated Command Control System (SACCS) current intelligence data base via the TOSS facility.

TRACALS-PIDP

Traffic Control and Landing Systems - Programmable Indicator Data Processor

An upgrade of USAF Radar Approach Control/Air Traffic Control (RAPCON/ATC) facilities to an automatic programmable capability compatible with the National Airspace System (NAS). The PIDP reduces air traffic controller radar scope aircraft tracking workloads and minimizes voice traffic among controllers, aircraft and adjacent control sites.

TRACALS-VFR Control Tower Simulator

Traffic Control and Landing Systems AN/GSN-T3

An electro-optical computer-driven large screen display and communications device. It is used by the Air Training Command (ATC) to train and exercise Air Force Communications Service (AFCS) control tower operators in Visual Flight Rules (VFR) procedures.

TRI-TAC

Triservice Combat Theater Communications

A program to define Air Force requirements for tactical ground communications, both near term and post-1980. It ensures that Air Force requirements are incorporated in the DoD Joint Tactical Communications Program (TRI-TAC). It also guarantees compability of Air Force developed equipment with similar apparatus procured by other agencies.

WFC

Wallops Flight Center, Wallops Island, Virginia

A National Aeronautics and Space Administration (NASA) facility which lofts several hundred experiments yearly on vehicles ranging from small meterological rockets to the four-stage Scout with orbital capacity. A sizable effort is devoted to aeronautical research and development.

Wild Weasel

An F-4C fighter aircraft configured in a defense suppression role. It carries electronic countermeasure (ECM) warning sensors, jamming pods, chaff dispensers, and anti-radiation missiles. Its avionics combine electronic warfare (EW) and navigation functions. The Advanced Wild Weasel is a modified F-4E aircraft with additional sophisticated EW equipment. It detects, identifies and locates enemy radars and then directs its weapons stores against them. Changing EW threats are covered by use of reprogrammable software.

WWMCCS/AFWWMCCS

Worldwide Military Command and Control System/Air Force Worldwide Military Command and Control System

A Department of Defense (DoD) system designed to link National Command Authorities (NCA) with commander of unified and specified field commands. It also supports command and control systems of those commands on the basis of non-interference with the Worldwide Military Command and Control System (WWMCCS) primary mission. WWMCCS components procured, owned and/or operated by the Air Force comprise the Air Force Worldwide Military Command and Control System (AFWWMCCS).

WWMCCS II

Worldwide Military Command and Control System II

The designation for the upgrade of Worldwide Military Command and Control System (WWMCCS) computers and automatic data processors (ADP).

REFERENCES

ARIN75 Aeronautical Radio, Inc.: "ARINC Report 416-10: Abbreviated Test Language for Avionics Systems (ATLAS)," Aeronautical Radio, Inc., Annapolis, Maryland, May 15, 1975.

AIRF71 Air Force, Department of the: "Data Automation: Computer Programming Languages," Air Force Regulation AFR 300-10, 20 October 1971.

AIRF67 Air Force, Department of the: "Standard Computer Programming Language for Air Force Command and Control Systems: Short Title: CED 2400," Air Force Manual AFM 100-24, 15 June 1967, (Change 1, 19 August 1969; Change 2, 9 Sept. 1969).

AIRF74 Air Force Department of the: "Automated Data Systems (ADS) Standards," AFM 171-100, Volume I: Machine Independent Standards, 1 November 1974, Volume II: H6000, 1 November 1975, Volume III: B3500 ADS, 1 September 1975.

ANSC76 American National Standards Committee X3J3: "Draft Proposed ANS FORTRAN BSR X3.9 X3J3/76," Sigplan Notices, Vol. II, No. 3 (March 1976).

ANSI66A American National Standards Institute: "American National Standard FORTRAN," ANS 3.9-1966, 1966.

ANSI66B American National Standards Institute: "American National Standard Basic FORTRAN," ANS X3.10-1966, 1966.

ANSI68 American National Standards Institute: "American National Standard Programming Language COBOL," ANSI X3.23-1968, 12 August 1968.

ANSI74 American National Standards Institute: "American National Standard Programming Language COBOL," ANSI X3.23-1974, May 10, 1975.

AUER76 Auerback, Issac (ed.): Auerbach Computer Technology Reports, Philadelphia, PA., Auerbach Information, Inc., 1976.

BBN 76 Bolt, Beranek and Newman, Inc.: "Development of a Communications Oriented Language," Part I AD A027519, Part II AD A027523, Cambridge, Mass., 20 March 1976.

CALL75 Callender, E. David, et al: "SAMSO Computer Language and Software Development Environment Requirements," AD-A018801, Aerospace Corp., El Segundo, Calif., 15 December 1975.

DATA76 Datapro Research Corp.: Datapro 70, Delcan, New Jersey, 1976.

DEPL75 Depledge, N. G.: "CORAL 66 - A Practical High Level Language for Minicomputer Software and Application Program Development," Honeywell Information Systems Ltd., Hemel Hempstead, England, April 1975.

DODD76 Department of Defense Directive 5000.29: "Management of Computer Resources in Major Defense Systems," 26 April 1976.

DODI76 Department of Defense Instruction 5000.31: "Interim List of DoD Approved High Order Programming Languages," 24 November 1976.

DREI76 Dreisbach, J. A., et al: "High Order Language Technology Evaluation Final Report," IR-203-1, Intermetrics, Inc., Cambridge, Mass., 15 October 1976.

DROU76 Drought, Mark (ed.): DMS Market Intelligence Reports, Los Angeles, California, DMS, Inc., 1976.

ENSL75 Enslow, Phillip H., Jr.: "CORAL 66: The UK Standard Programming Language for Weapons Systems," in USAR75, 10 April 1975, 3-1 to 3-42.

FALK76 Falk, Harold C.: "AFSC Higher Order Language Standardization Program: Information from System Program Offices ASD--Operational Flight Program Area," WPAFB, Engineering Report ASD/ENA-76-7, 11 March 1976.

FELT76 Felty, J. L., Roth, M. S.: "High Order Language Technology Evaluation: Software Support Tools," IR-204-1, Intermetrics, Inc., Cambrdige, Mass., 15 October 1976.

FISH76 Fisher, David A.: "A Common Programming Language for the Department of Defense -- Background and Technical Requirements," Institute for Defense Analysis, Arlington, Virginia, P-1191, AD A028 297, June 1976.

GML GML Corp.: Minicomputer Review, Lexington, Mass., Jan. 1976.

LAPA76 La Padula, L. J. and P. L. Loring, "Air Force Programming Languages: Standards, Use, and Selection," ESD-TR-76-140, Electronic Systems Division, AFSC, Hanscom AF Base, Mass., August 1976.

MART74 Martin, Fred H.: "HAL/S: The Programming Language for Shuttle," Intermetrics, Inc., Report IR-92-1, Cambridge, Mass., 4 October 1974.

MART75 Martin, Fred H.: "On the Performance of the HAL/S-FC Compiler," Intermetrics, Inc., IR-162, Cambridge, Mass., 22 October 1975.

NEVE74 Neve, N. J. F. et al: "A Set of Programs to Assist in the Assessment of CORAL 66 Compilers," Royal Radar Establishment Tech Note No. 782, AD-A004917, May 1974.

RADC75 Rome Air Development Center, "Structured Programming Series," International Business Machines, RADC-TR-74-300, Vol. I to Vol. XV, 1974-1975.

RADC76 Rome Air Development Center: "JOVIAL J73/I Specification," RADC, Air Force Systems Command, Griffiss Air Force Base, N.Y., July 1976.

SOFT76A Softech, Inc.: "High Order Language (HOL) Investigation," Technical Proposal 8469, Waltham, Mass., 4 March 1976.

SOFT76B Softech, Inc.: "JOVIAL/J3B Language Specification Extension 2," Report 244-4.3, Waltham, Mass., August 1976.

TRAI76 Trainor, Lynn: "Interim Report on the Status of the JOVIAL-73 and JOVIAL-3B Languages," Air Force Avionics Lab, WPAFB, Ohio, August 1976.

USAR75 U.S. Army Research and Development Group (Europe): "Implementation Languages and Real-Time Systems," European Research Office (England), Tech Report No. ERO-2-75, Vol. 1, 2, 3, 15 April 1975.

WOOD73 Woodward, P. M., et al: "Official Definition of CORAL 66," Ministry of Defence, Royal Radar Establishment, Malvern, England, February 1973.